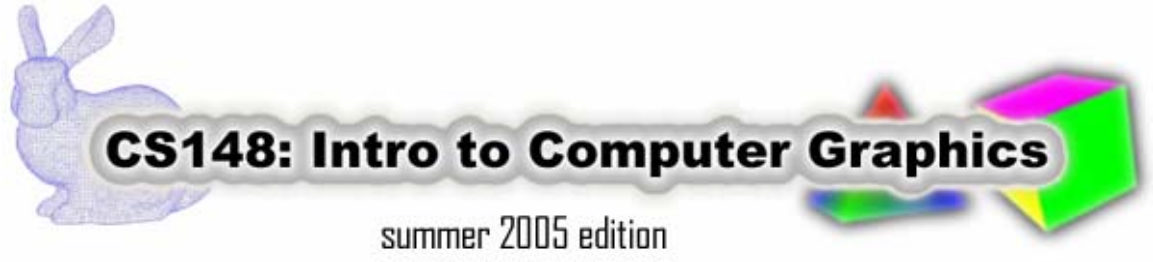


Handout #1: Information and Syllabus

Instructor Dan Morris
Gates 116
cs148instructor@cs.stanford.edu
Office Hours: Tuesdays, 1 pm to 3 pm, Gates 116, or by appointment

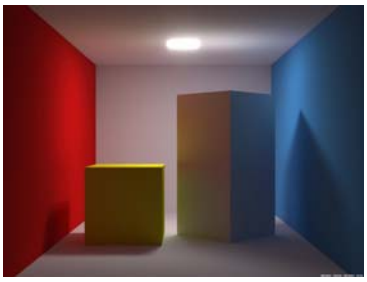
TA Sean Walker
cs148ta@cs.stanford.edu
Office Hours: TBD

Contact Information

The course staff may be reached at cs148staff@cs.stanford.edu – using this list will often be faster than e-mailing an individual staff member. Remember that the course staff is human and we aren't always at a computer – we will try to respond within 24 hours (hopefully sooner). A newsgroup has also been set up at su.class.cs148 for discussion and student help.

The class website is http://cs148.stanford.edu. Try to check it at least twice a week so you don't miss important announcements.

CS148: Introductory Graphics



This course is an introduction to 2-dimensional and 3-dimensional computer graphics, with an emphasis on interactive 3D graphics. Topics covered include scan conversion, OpenGL and GLUT, matrix transformations and clipping, graphical model representations, projection algorithms, lighting and color models, hidden-surface elimination, Bezier and B-Spline curves, animation, and ray tracing.

There are two main goals of CS148:

- Theoretical: To introduce important concepts in the field of graphics
- Practical: To give you a working knowledge of OpenGL

Note: This is an undergraduate terminal course. Students with a strong interest in continuing in graphics should take CS248 as their introductory course.

Pre-requisites

CS107 (Programming Paradigms), Math 103 (Linear Algebra)

There are no strict rules about prerequisites; talk to Dan if you're not sure whether you're prepared for cs148.

Lectures

TTh, 11:00 am to 12:50 pm, Gates B01
SITN: Channel E4, live
SCPD: Available online about two hours after class

Assignments & Exams

There will be an in-class midterm, a final, and four programming projects. We need to be notified about exam conflicts two weeks in advance of the exam in order to schedule an alternate exam time.

We welcome exam question suggestions from students, and if you think of a good question (and answer), we might just use it. No specific extra credit will be given, but there's nothing more relaxing than seeing your own question on an exam. :) (Please note that submitting plagiarized exam questions will be considered an honor code violation and will be treated just like cheating.)

All assignments are due by 11:59 p.m. on the dates specified in the syllabus. Early assignments will be done individually while the last two assignments, which are more open-ended, will have an option for working in pairs or groups of three. We expect the work of a group to be proportionally more complicated than an individual effort. If you choose to work with others, be sure that all of you work on the entire assignment together. That is, we cannot accept "half" of an assignment from you if your partner failed to do his/her part.

Information on how to submit projects will be provided with the project handouts.

Late Policy

At the beginning of the quarter, each student has 3 "late days" that can be used to extend project due dates. A late day is considered a 24 hour period – for example, if the assignment is due at 11:59 pm and you hand it in at 3:12 am, you will have used 1 "late day" (since we round up to the nearest day) and you will have 2 days left. Days may all be used on a single assignment or be split across multiple assignments. After late days have been used up, we will penalize 25% of the assignment value for each day that the assignment is late (rounded up to the next largest 24 hour period). So, let's say you have 1 late day left and you hand in an assignment 25 hours late – even though you are only an hour over, we would still penalize you 25% because we round up to the next whole day. If you are working with a partner, you will both lose late days if you hand in an assignment late.

You should e-mail the TA before or on the due date if you will be handing in an assignment late – this will prevent the TA from grading any previous submission you have made. You should also e-mail the TA again when your final (late) submission is in place to let her know it can be graded.

Extensions for good reasons may be granted if they are requested prior to the due date. All such requests must be made directly to the instructor, not the TA.

Grading

Final grades will be based on the following:

50% Programming Projects
20% Midterm
30% Final

To receive a passing grade, you must complete *passing work* on both exams and all projects (before any late penalties).

Sufficiently justified regrades may be requested within one week after an exam or project is handed back.

Textbooks

The official textbook for CS148 is the *OpenGL Programming Guide v1.4, Fourth Edition* ("The Red Book") by Shreiner et al. While the class will be based on the fourth edition, the second edition (covering OpenGL v1.1) will be sufficient for the class since we will rarely use the advanced features. The second edition can be found online; a link to it is available on our website. We also provide a link to an online OpenGL

API reference.

The textbook will primarily be used as a reference for your programming projects; the majority of the course material will be presented in class. Lecture slides will generally be posted on the website the night before class – it is the students' responsibility to print a copy if they want to refer to it during class.

The reading assignments in the syllabus are approximate – since the *OpenGL Programming Guide* focuses only on OpenGL it does not follow the same order as this class. The timing of the readings is designed to give the best fit with the material covered in this course. However, you may find (depending on your programming and math knowledge) that you need to read more background material or the previous chapters to understand what is in the reading assignment.

Programming Environment

OpenGL is a major component of CS148 and it will be used for all of the projects in the class. We will also be using GLUT (the OpenGL Utility Toolkit) which abstracts the window system calls and simplifies development greatly.

While OpenGL and GLUT are supported on many platforms, we will test and grade your programs on the *myth, firebird,* and *raptor* machines in Sweet Hall so **your programs must run in Linux**. It is fine if you want to develop on your home machine under Windows or another operating system but be sure to verify that each of your solutions works on the Linux machines in Sweet Hall.

Instructions for developing in OpenGL and GLUT will be distributed with the first programming project.

How to Succeed

CS148 is a programming-intensive course. Be aware that you will be doing a lot of coding – there will be a project due just about every week, so you need to budget your time accordingly.

Extra credit will be given for projects that go above and beyond the assignment requirements. A computer graphics course is a great opportunity to be creative and make pretty pictures, and we'll give extra credit for particularly creative projects or projects that make use of advanced OpenGL features.

To succeed in this class:

1. Make note of when projects will be distributed (see below) and make sure you allot enough time to complete them.
2. Come to class… we will run through numerous OpenGL demos and exercises, and we'll try to have a good time doing it.
3. Your projects are graded for the most part, on output only. We expect fully functioning programs that fulfill the requirements as stated in the project handouts. If something does not compile or crashes, the TA will not spend much time on grading it.
4. Remember that the TA is not a debugger: debugging is a useful skill for you to learn, so please don't expect the TA to debug your program in office hours.
5. Take advantage of extra-credit – it's a good way to improve your grade if you make a few mistakes.

Honor Code

Programming projects are to be implemented "from scratch", i.e., it is a violation of the honor code to derive solutions from existing sources or previous instances of this course. We reserve the right to use MOSS or other plagiarism-detection software on your submissions. Discussion of programming projects at a high conceptual level is allowed. Copying of solutions from other students, textbooks, or from students who previously took this course is not allowed. If you do discuss a project with another student, you must document this in your program. A good guideline is that you must be able to explain and/or fully duplicate anything that you submit.

Disability Statement

Students who have a disability which may necessitate an academic accommodation or the use of auxiliary aids and services in a class must initiate the request with the Disability Resource Center (DRC)**.** The DRC will evaluate the request with required documentation, recommend appropriate accommodations, and prepare a verification letter dated in the current academic term in which the request is being made. Please contact the DRC as soon as possible; timely notice is needed to arrange for appropriate accommodations. The DRC is located at 123 Meyer Library (phone 723-1066; TDD 723-1067).

Special Thanks

Special thanks to Sean – your hard-working TA – who taught the course previously and has provided his lecture notes and slides as a basis for much of what you'll see this quarter. His notes, in turn, were derived from Maggie Johnson's notes, so we thank her here too.

CS148 Syllabus – Summer 2005

Date	Topic	Readings
6/21 (Tues)	CS148 overview Required math Display devices PROJECT 1 GOES OUT	
6/23 (Thurs)	Scan Conversion	Essential Math Handout
6/28 (Tues)	Intro to OpenGL Windows and clipping	Chapters 1 & 2
6/30 (Thurs)	Animation and UI Display lists Vertex Arrays	Chapter 7 Appendix D
7/5 (Tues)	Transformations (2D & 3D) The depth buffer The matrix stack PROJECT 1 DUE PROJECT 2 GOES OUT	Chapter 3 Appendix F
7/7 (Thurs)	3D viewing	
7/12 (Tues)	Modeling objects Quadrics in OpenGL Display lists	Chapter 11 Appendix E
7/14 (Thurs)	Meshes	
7/19 (Tues)	Midterm PROJECT 2 DUE PROJECT 3 GOES OUT	
7/21 (Thurs)	Lighting and shading	Chapter 5
7/26 (Tues)	Color and texture	Chapters 4 & 9
7/28 (Thurs)	Curves and curved surfaces PROJECT 4 GOES OUT PROJECT 3 DUE	Chapter 12
8/2 (Tues)	Raytracing OpenGL selection	Chapter 13
8/4 (Thurs)	Transparency in OpenGL Hidden surface elimination	Chapter 10 Chapter 6
8/9 (Tues)	Physical simulation Game development	
8/11 (Thursday)	Review / Hot topics in graphics / TBD PROJECT 4 DUE	
8/12 (Friday)	Final: location TBD	