

- Files
- master
- Go to file
- images
- lectures
- tutorials
- .gitignore
- README.md
- github-markdown.css
- header.md
- lecture2.pdf

computer-graphics-csc317 / README.md

dilevin Update README.md 117d91d · now History

computer-graphics-csc317 / README.md Preview Code Blame 165 lines (119 loc) · 6.49 KB Raw

# Computer Graphics CSC317/CSC2504 Fall 2023



- Lectures
- Tutorials
- Course Overview
- Required Textbook
- Lecture Schedule
- Grading
- Lateness Policy

## Instructors

Prof. David I.W. Levin  
diwlevin@cs.toronto.edu  
+1 416-978-2052

## TAs

Jonathan Panuelos  
Wenzhi Guo  
Zhecheng Wang  
csc317tas@cs.toronto.edu

Office Hours: Friday 1-2pm in BA5268

## Lectures

Mondays 11:00-13:00 in FE230

## Tutorials

Wednesday 11:00-12:00 in UC140

## Links

- MarkUs

Use Assignment GitHub issue pages for questions/discussion

## Programming Resources

- www.cplusplus.com
- OpenGL Tutorials at <http://www.opengl-tutorial.org>
- The Matrix Cookbook

## Additional Graphics Resources (papers/presentations)

\*\* You need to login to the UofT vpn to access these \*\*

- [Simulating characters in vehicles for games](#)
- [Shading Cloth](#)

## Course Overview

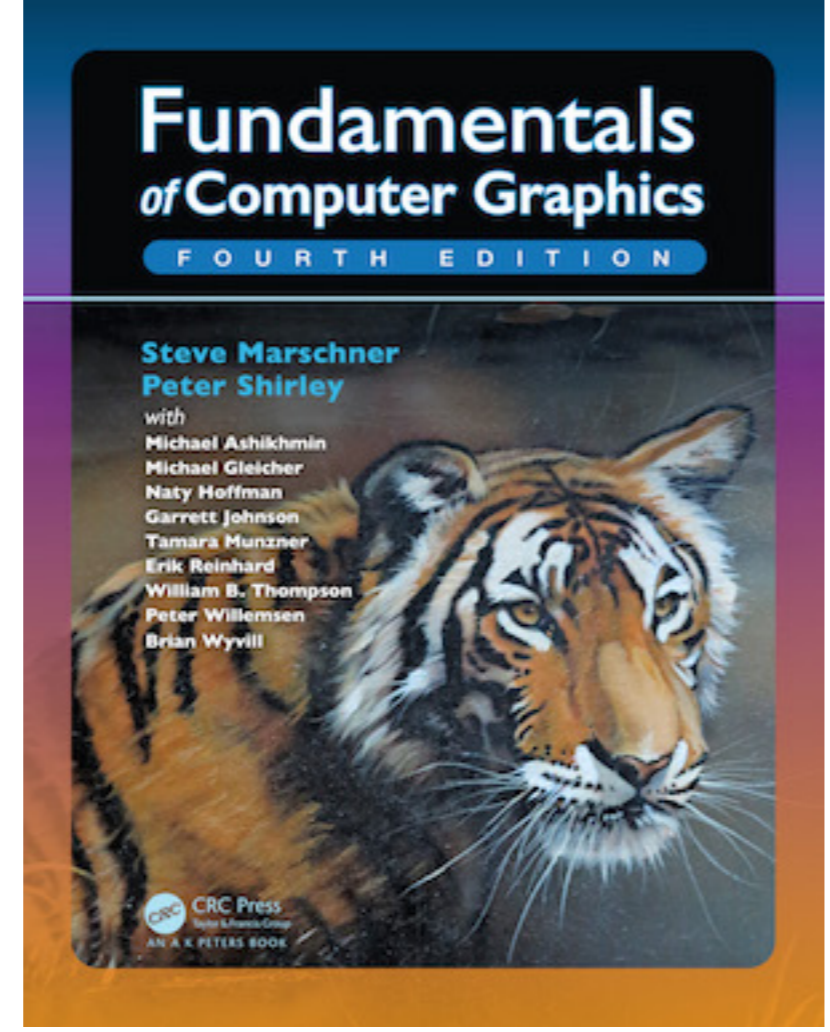
This course introduces the basic concepts and algorithms of computer graphics. It covers the basic methods needed to model and render 3D objects, including much of the following: graphics displays, basic optics, line drawing, affine and perspective transformations, windows and viewports, clipping, visibility, illumination and reflectance models, radiometry, energy transfer models, parametric representations, curves and surfaces, texture mapping, graphics hardware, ray tracing, graphics toolkits, animation systems.

Prerequisites: C/C++ Programming, Linear Algebra, Calculus, Numerical Methods.

The student is expected to read background material on the hardware and local software, and should be comfortable with elementary linear algebra, geometry, and vector calculus. It is also assumed that the student is comfortable programming in basic C++.

(Strongly) Recommended preparation: Multivariable Calculus

## Required Textbook



This class involves *required reading* from:

*Fundamentals of Computer Graphics, Fourth Edition*, Steve Marschner, Peter Shirley, et al. 2015.

Digital e-book are available at [CRC Press](#).

## Lecture Schedule

Week	Topic / Event
1	Introduction, tutorial, Assignment 1 (Raster Images) waitlisted ? zip assignment and email to TAs due 20/09
2	Lecture 2, Assignment 2 Ray Casting due 27/09
3	Lecture 3, Assignment 3 Ray Tracing due 04/10
4	Lecture 4, Assignment 4 Boundary Volume Hierarchy due 13/10
	24 hour Take home midterm (Exact date to be determined)
5	No Lecture, Thanksgiving
6	Lecture 5, Assignment 5 Meshes due 25/10
7	Lecture 6, Assignment 6 Shader Pipeline due 01/11
8	Lecture 7, Assignment 7 Kinematics due 15/11
9	No Lecture, Reading Week !
10	Lecture 8, Assignment 8 Mass-Spring Systems due 22/11
11	Lecture 9, Assignment on Generative AI due 29/11
12	Lecture 10 New Developments in Computer Graphics

[Academic Honesty \(required reading\)](#)



## Grading

%	Item
72	Assignments
14	Midterm
14	In Tutorial Final Exam

There will be nine programming assignments in total (8% each).

## Lateness Policy

Assignments are *due by 11:59pm* on the due date.

0.007% off for every minute late.

## Academic Honesty

Academic honesty is a very serious matter and can result in very serious consequences. Note that academic offences may be discovered and handled retroactively, even after the semester in which the course was taken for credit. This is a challenging class aimed at teaching you the fundamentals of computer graphics. You wont learn much if you cheat but you might get a good grade if you get away with it. If all you want is a good grade take an easier class where you wont have to cheat!

For purposes of this class, academic dishonesty is defined as:

- Any attempt to pass off work on a test that didn't come straight out of your own head.
- Any collaboration on written or programming assignments (its ok to share ideas on programming assignments but the code MUST be your own) in which the collaborating parties don't clearly and prominently explain exactly who did what, at turn-in time.
- Any activity that has the effect of significantly impairing the ability of another student to learn. Examples here might include destroying the work of others, interfering with their access to resources (e.g., digital cameras), or deliberately providing them with misleading information.

## Email & Bulletin Board Traffic

- Please do not send email directly to the TAs. They will not be replied.
- Main forum for answering questions about class or about the assignments is the class bulletin board. The TAs will be monitoring the board.
- Appropriate use of the board: clarifications on assignment, on lecture material, general concerns about the course, or other remarks that are appropriate for all students to see/participate in.
- Do NOT broadcast pieces of your code or answers to written assignments to the bulletin board. Specific or general implementation questions whose answer would benefit all students are appropriate. However: the bulletin board is NO replacement for the tutorial hour. That should be the main forum for asking/answering questions of this sort.
- Questions of the form "I cannot find the problem with my code; here it is, can you help me" are unlikely to be replied, so don't count on it. If you have a question with code, take it to the TA office hours or to the tutorials.