Syllabus, CSCI 480, Computer Graphics, Fall 2014

Instructor: Geoffrey Matthews, x3797
Office hours: MTWF 9:00, CF 469
Email: geoffrey dot matthews at wwu dot edu

Websites:

- For class materials: `https://github.com/geofmatthews/csci480`
- For turning in homework and grading: `https://wwu.instructure.com/`

Lectures: MTWF 10:00-10:50, MH 038

Goals: This class is an introduction to fundamental algorithms of computer graphics, and to programming in OpenGL. We will implement several algorithms for the generation of images in a general purpose programming language, and then we will study how those methods, and others, have been implemented in hardware in the OpenGL pipeline.

By the end of this course the student should be able to:

- Use basic linear algebra in applications to computer graphics.
- Use homogeneous coordinates to represent points and vectors.
- Produce and use geometric transformations represented as matrices.
- Understand the spaces and projections used in 3D graphics.
- Understand the basic difference between physically based rendering and direct rendering, and program each kind of renderer.
- Program graphical applications in 3D using OpenGL.
- Understand the OpenGL pipeline.
- Understand vertex and fragment shaders, and program them in GLSL, the OpenGL Shader Language.

Text: Online notes and web pages will be used. I have assembled relevant wikipedia articles into a book that covers most of what we need:
`https://en.wikipedia.org/wiki/Book:CSCI_480_Computer_Graphics`

Homework: Usually paper&pencil math type homework, as assigned through the quarter.

Programs: As assigned through the quarter. Programming language for the course is python.

Exams: One midterm and one final, as on the schedule below.

Exams are closed book, with the exception that you may consult two pieces of paper during the exam. You may write or print whatever you wish on these pages.

Grading:

Homework	Programs	Midterm	Final
25%	25%	20%	30%

Letter grades:

$A \geq 90\% > B \geq 80\% > C \geq 70\% > D \geq 60\% > F$

Plus and minus is at the discretion of the instructor.

Special projects: Students may optionally complete one extra credit project for a maximum of 10 percentage points of extra credit. Special projects must: (a) be proposed to the instructor, in writing, at least three weeks before the last day of classes, (b) be approved by the instructor at that time, (c) must include a substantial programming component, (d) must include a writeup with an introduction

outlining the purpose of the project, a section discussing the results, conclusions, *etc.*, (e) must include well annotated source code, a user's guide, and a programmer's guide to the software, and (f) must be complete and turned in before the last week of class. There are no other extra credit opportunities for this class.

Academic dishonesty: Academic dishonesty policy and procedure is discussed in the University Catalog, Appendix D. All students should read this section of the catalog. It consists of misrepresentation by deception or other fraudulent means. In computer science courses this frequently takes the form of copying another's program, either a fellow student's program, or copying one from the web. Due diligence should be exercised in the labs at all times, since both copying and letting someone else copy your program are equally culpable. Do not walk away from your computer in the lab without logging out or locking the screen. Do not share files, even if it is just to "show them something." Describe it in words, or talk to them in person; never share code.

Collaboration: Collaboration with your fellow students is a good way to learn. Feel free to share ideas, solve problems, and discuss your programs with other students. However, collaboration is *not* copying. All code should be original. Remember **the Simpsons Rule:** after discussing homework with another student, you must erase the board and destroy all written notes, pictures, files, *etc.* that you shared. After that, you must watch a rerun of *the Simpsons*, or do something else unrelated, for half an hour. Then you can take the knowledge you gained from another student and put it to work, since it is now not copying, but learning. You have made it your own.

Schedule: rough guide to the weeks ahead:

- Week 1: Introduction, python, pygame
- Week 2: noise, linear algebra
- Week 3-4: raytracing: intersection tests, Phong reflection model, normals, Whitted raytracing, texture coordinates, implicit surfaces, parametric surfaces, derivatives,
- Week 5: transforms, direct rendering, line rendering, polygon filling, Phong shading
- Wednesday, October 28: Midterm
- Weeks 6-10: OpenGL
- Tuesday, Dec 8: Final exam, 8:00am.