

COGS 4550 Game Architecture

Mondays, Thursdays 2PM - 3:50PM

Sage 2510

Instructor: Ben Chang

Office: West Hall 104

Email: changb3@rpi.edu

Office Hours: Tuesdays 1-3

Book: *Game Engine Architecture, 3rd edition*, by Jason Gregory (recommended)

Course Description:

This is a class on the composition of game engines, some of the most complex software applications that exist. We will discuss many features common to game engines, such as collision detection, event management, and scene graph manipulations. The first two-thirds of the class will be primarily lecture-based, covering topics needed for all game engines. The last third of the class will be primarily project-based, with individuals or teams focusing on an aspect of game architecture that they would like to expand further.

Learning Outcomes:

Students will learn a more detailed understanding of the subsystems that comprise a game engine, and how they interrelate. Each student in the class will either create a minimalist game engine from scratch, or make a notable contribution to an existing open-source game engine.

Academic Honesty:

Student relationships are built on mutual respect and trust. Students must be able to trust that their teachers have made responsible decisions about the structure and content of the course and that they are conscientiously making their best effort to help students learn. Teachers must be able to trust that students do their work conscientiously and honestly making their best effort to learn. Acts that violate this mutual respect and trust undermine the educational process. They counteract our very reason for being here at Rensselaer and will not be tolerated. Any student who engages in any form of academic dishonesty will receive an F in this course and will be reported to the Dean of Students for further disciplinary action. The Rensselaer Handbook defined various forms of Academic Dishonesty and procedures for responding to them. All of these forms are violations of trust between students and teachers. Please familiarize yourself with this portion of the handbook.

Programming is rarely a purely solitary endeavor, but for the sake of academic integrity, it is essential that you do your own work. There is a reasonable line between asking people for help, and outright plagiarism. You may ask for help,

and you may show others your code, but you may not view their code. You may help other people, but you may not show them your own work. Don't guilt-trip people, and don't let yourself be guilt-tripped. Friends don't manipulate each other.

You MAY:

- Whiteboard solutions with others using diagrams or pseudocode but not actual code.
- Use the web or elsewhere for instruction beyond the course's own, for references, and for solutions to technical difficulties, but not for outright solutions to homeworks.
- Incorporate snippets of code (2 lines as a guideline) that you find online or elsewhere into your own code, provided that those snippets are not themselves solutions to assigned problems and that you cite its origins.
- Run someone else's code on your computer.

You may NOT:

- Ask a classmate to see their solution to a problem before submitting your own.
- Give or show a classmate a solution to a problem when they are struggling to solve it.
- Search for or solicit outright solutions to problem online or elsewhere.
- Split a problem's workload with another individual and combining your work.
- Submit the same or similar work to this course that you have submitted or will submit to another.
- Splitting a problem set's workload with another individual and then combine your work.
- View another's solution to a problem and then base your own solution on it.

Grading: The graded assignments for this class will include six homework assignments, each of which will expand our bare-bones engine. The final project will be a software project kicked off by a pitch-style presentation. Your final grade is determined as follows:

HW 1-6: 12.5% each

Final Project: 25%

Week	Date	Topic	Reading	Assignment Due
1	1/10	Introduction	Ch 1-3	
2	1/14	Entity Systems, ga_framework codebase intro		
	1/17	Main Loop		
3	1/21	No class, Martin Luther King Day		HW 1 Due
	1/24	Debugging	Ch 9	
4	1/28	Math (Vectors, Matrices)	Ch 4	
	1/31	Math (Vectors, Matrices)		
5	2/4	Graphics	Ch 10	HW 2 Due
	2/7	Graphics		
6	2/11	Graphics		
	2/14	Graphics		
7	2/18	No Class, Presidents' Day		HW 3 Due
	2/21	GUI		
8	2/25	Collision Detection	Ch 12	HW 4 Due
	2/28	Rigid Body Dynamics		
9	3/4	No Class, Spring Break		
	3/7	No Class, Spring Break		
10	3/11	Animation	Ch 11	HW 5 Due
	3/14	Animation		
11	3/18	GDC - guest lecture TBA		
	3/21	GDC - guest lecture TBA		
12	3/25	Runtime scripting	Ch 14	HW 6 Due
	3/28	Audio		
13	4/1	Networking		HW 7 Due
	4/4	Final Project Pitch (in-class)		
14	4/8	Studio		
	4/11	Studio		
15	4/15	Studio		
	4/18	Studio		
16	4/22	Studio		
	4/25	Final Project Presentations		Final Project Due