USC Viterbi
School of Engineering

Programming Game Engines

ITP 485 (4 Units)
Spring 2017

Objective

This course provides students with an in-depth exploration of 3D game engine architecture.

Students will learn state-of-the-art software architecture principles in the context of game engine design, investigate subsystems typically found in a real game engine, survey engine architectures from actual shipped games, and explore how the differences between game genres can affect engine design.

Students will participate in individual hands-on lab exercises to reinforce these concepts.

Concepts

Engine subsystems including rendering, audio, collision, physics, and game world models. Large-scale C++ software architecture in a games context. Tools pipelines for modern games.

Prerequisites

ITP 380

Instructors

Matt Whiting

Email

Email: whitingm@usc.edu

Office Hours

Matt Whiting: Email me and we'll set up a skype
Michael Feng: ?
Thomas Wilson: ?

TA

Michael Feng: fengm@usc.edu
Thomas Wilson: ?

Time/Location

6-8pm KAP 163

Course Structure

Throughout the semester, students will work by themselves to build features in a skeleton (or "toy") game engine. These assignments must be completed *individually*.

From time to time during the semester, we'll have in-class assignments. Each in-class assignment is to be completed individually during the time allotted during that class period and is "open-book". Any and all reference material is allowed, but collaboration is not. This is a chance to practice finding and using reference material.

There are two exams which are comprehensive of all topics covered.

Textbooks

Required:
Game Engine Architecture, Second Edition. Jason Gregory. ISBN-13: 978-1466560017.
Optional:
Effective C++ (3rd Edition). Scott Meyers. ISBN-13: 978-0321334879.

Grading

The course is graded with the following weights:

Lab Assignments	30%
In-Class Assignments	20%
Midterm Exam	20%
Final Exam	30%
TOTAL POSSIBLE	100%

Grading Scale

Letter grades will be assigned according to the following scale:

93%+	A
90-92%	A-
87-89%	B+
83-86%	B
80-82%	B-
77-79%	C+
73-76%	C
70-72%	C-
69	D+
67-68	D
66	D-
65 and below	F

Half percentage points will be rounded up to the next whole percentage. So for instance, 89.5% is an A-, but 89.4% is a B+.

There is no curving. Students will receive the grade they earn. Extra credit is generally not offered.

Policies

Make-up policy for exams: To make up for a missed exam, the student must provide a satisfactory reason (as determined by the instructor) along with proper documentation. Make-up exams are only allowed under extraordinary and emergency circumstances.

Late Lab Assignments: Lab assignments will be accepted late with a 10% penalty per day late, up to three days late. Assignments more than three days late will not be accepted.

Software

Due to the nature of programming with the DirectX API, students should have access to a machine with Windows. If you are on a Mac, you can download Windows from USC Viterbi Dreamspark, and install it on your Mac via Bootcamp. Because we are using DirectX 11, Parallels or VMWare fusion do not work. You have to boot via bootcamp.

Statement on Academic Conduct and Support Systems

Academic Conduct

Plagiarism – presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in *SCampus* in Section 11, *Behavior Violating University Standards* https://scampus.usc.edu/1100-behavior-violating-university-standards-and-appropriate-sanctions/. Other forms of academic dishonesty are equally unacceptable. See additional information in *SCampus* and university policies on scientific misconduct, http://policy.usc.edu/scientific-misconduct/.

Discrimination, sexual assault, and harassment are not tolerated by the university. You are encouraged to report any incidents to the *Office of Equity and Diversity* http://equity.usc.edu/ or to the *Department of Public Safety* http://capsnet.usc.edu/department/department-public-safety/online-forms/contact-us. This is important for the safety whole USC community. Another member of the university community – such as a friend, classmate, advisor, or faculty member – can help initiate the report, or can initiate the report on behalf of another person. *The Center for Women and Men* http://www.usc.edu/student-affairs/cwm/ provides 24/7 confidential support, and the sexual assault resource center webpage sarc.usc.edu describes reporting options and other resources.

Support Systems

A number of USC's schools provide support for students who need help with scholarly writing. Check with your advisor or program staff to find out more. Students whose primary language is not English should check with the *American Language Institute* http://dornsife.usc.edu/ali, which sponsors courses and workshops specifically for international graduate students. *The Office of Disability Services and Programs* http://sait.usc.edu/academicsupport/centerprograms/dsp/home_index.html provides certification for students with disabilities and helps arrange the relevant accommodations. If an officially declared emergency makes travel to campus infeasible, *USC Emergency Information* *http://emergency.usc.edu/* will provide safety and other updates, including ways in which instruction will be continued by means of blackboard, teleconferencing, and other technology.

A Further Note on Plagiarism

In this class, all homework submissions will be compared with current, previous, and future students' submissions using MOSS, which is a code plagiarism identification program. If your code significantly matches another student's submission, you will be reported to SJACS with the recommended penalty of an F in the course.

It is okay to discuss solutions to specific problems with other students, but it is not okay to look through another student's code. It does not matter if this code is online or from a student you know, it is cheating. Do not share your code with anyone else in this or a future section of the course, as allowing someone else to copy your code carries the same penalty as you copying the code yourself.

Course Outline

W		Date	Topic(s)	Reading/Due Dates
1	1	1/10	Introduction; SIMD	§4.7; Blackboard “SIMD Tutorial”;
	2	1/12	*Custom Memory Allocators; Utilities	§5.2; §3.3
2	3	1/17	Lab Session – Finish Lab 1 (SIMD)	**Lab 1 DUE 1/18 @ 11:59PM**
	4	1/19	Math Review; Rendering and Shaders	§10.1 – §10.2
3	5	1/24	*Intermediate Rendering I	§10.3 – §10.5
	6	1/26	Intermediate Rendering II	
4	7	1/31	Lab Session – Finish Lab 2 (Graphics 1)	**Lab 2 DUE 2/1 @ 11:59PM**
	8	2/2	*Rendering III	
5	9	2/7	The C++ Compiler	
	10	2/9	*Game Object Models; Data Management	§15.1 – §15.4;
6	11	2/14	Caching and Performance	§3.4, §2.3, §9.8
	12	2/16	Lab Session – Finish Lab 3 (Graphics 2)	**Lab 3 DUE 2/17 @ 11:59PM**
7	13	2/21	*Midterm Review	
	14	2/23	**Midterm Exam**	
8	15	2/28	Animation System Architecture	§11.1 – §11.10
	16	3/2	Hardware; 3D Math	§4.1 – §4.6; §4.8
9	17	3/7	*Multithreading	§7.6; §15.6;
	18	3/9	Lab Session – Finish Lab 4 (Animation)	**Lab 4 DUE 3/10 @ 11:59PM**
10		3/14	Spring Recess	
		3/16	Spring Recess	
11	19	3/21	Collision Detection; Physics Engines	§12.3; §12.5
	20	3/23	*Audio Systems	§13.1 – §13.6;
12	21	3/28	Lab Session – Finish Lab 5 (Collisions)	**Lab 5 DUE 3/29 @ 11:59PM**
	22	3/30	Multiplayer	§7.7
13	23	4/4	Lab Session – Finish Lab 6 (Normal Map)	**Lab 6 DUE 4/5 @ 11:59PM**
	24	4/6	Content/Engine Pipelines	§14.4
14	25	4/11	Lab Session – Finish Lab 7 (Profiling)	**Lab 7 DUE 4/12 @ 11:59PM**
	26	4/13	Scripting and Gameplay Foundations (Guest Lecture – Jason Gregory)	§15.7 - §15.9
15	27	4/18	Lab Session – Finish Lab 8 (Post Effects)	**Lab 8 DUE 4/19 @ 11:59PM**
	28	4/20	TDB	
16	29	4/25	Lab Session – Finish Lab 9 (Multithreading)	**Lab 9 DUE 4/26 @ 11:59PM**
	30	4/27	Final Review	
17		5/4	**Final Exam 7 - 9 pm**	