

Game AI

Course numbers: CS 4731 / CS 7632

Monday/Wednesday/Friday 11:05 - 11:55

Howey (Physics) L3

<http://www.cc.gatech.edu/~riedl/classes/2017/gameai/>

Instructor: Mark Riedl, riedl@cc.gatech.edu

228 Technology Square Research Building (TSRB)

Office Hours: Thursdays 2:30-4:30pm, and by appointment

Overview

The purpose of this course is for undergraduates and graduates in Computer Science and Computational Media to gain a breadth of understanding in the toolbox of AI approaches employed in digital games. This involves learning some basic topics covered in other AI courses, but with a focus on applied knowledge within the context of digital games.

Game AI is distinct from "academic AI" in that the end behavior is the target. Game AI programmers are less concerned with the underlying algorithms and more so with the end result. For example, if having an AI 'cheat' provides a more entertaining experience, than cheating will likely be a main component of the design. There are also characteristics of many games that focus Game AI on specific problems, like navigation through a virtual world, tactics, and believable behavior. Academic AI researchers are more concerned with rational behavior, knowledge representations, robust multi-agent communication, etc. However, there are overlaps between the two domains, where the desired behavior requires less cheating and more realistic decision-making. This course will survey topics related to this overlap, with a focus on applying what we review in depth through implementations in digital games

This course also observes the difference between AI as a technical challenge for opposing forces AI in games and the integration of AI as a key aesthetic component of the gaming experience. Lectures and projects will explore both of these views of Game AI.

This syllabus should be considered a living document subject to change throughout the course of the semester. There are multiple places in the class schedule to accommodate student interests in particular subjects.

Expected Outcomes

At the conclusion of this class, students should be capable of implementing some of the most common Game AI techniques used in industry today. Students will be familiar with a range of techniques being used to control agent behaviors in modern computer game and to procedurally generate content for computer games. Students will have an understanding of and appreciate the

distinctions between artificial intelligence techniques adopted by the computer game industry and that being pursued in research labs and non-games related industries.

Requirements

Students should have taken a previous course on artificial intelligence. For Georgia Tech undergraduates, this is CS 3600 "Introduction to Artificial Intelligence". For graduate students this may be CS 6601 or any equivalent course from their undergrad institution. At a minimum, the course should have covered search algorithms such as A* and hill-climbing search and some machine learning.

Students are required to have solid programming skills. Experience with Java or the ability to pick it up as part of the course is required. Students are expected to be able to pick up pre-existing code bases and develop their AI code within that code base as part of the class.

Course Structure

Lectures: The course will be conducted through lectures and occasional participatory exercises.

Homeworks: Students will demonstrate their proficiency at Game AI material through a sequence of short-sprint (1-2 week) homework assignments. The sequence of homework assignments together comprises a full-scale project in which all aspects of the AI for complete game are implemented. Because homework assignments build on each other, the instructor will provide sample solutions after each sprint.

Exams: There will be two exams throughout the semester.

Course Communication: We will use Piazza as our main method of electronic communications and announcements. All students should join the "CS 4731" course. Students will be responsible for any announcements made there. The use of the group is a resource for technical and design issues. For assignment submissions, we will use T-Square.

Grading

There are 8 homework assignments. Your final grade will be computed as follows:

- Homework assignments (8): 70% total
- Exams (2): 15% each

There may be one or two optional homework assignments for extra credit toward one's final course grade. At least one of the extra credit projects will be to participate in a bot-versus-bot competition.

Additionally, there may be opportunities to participate in experiments run by graduate students performing AI research. I will announce opportunities as I become aware of them. Meeting certain criteria as a participant can result in additional 2 extra points added to your final grade. This cannot be used more than twice, although you may participate in as many experiments as you wish. See extra credit policy on the website for more information.

Honor Code

I reserve the right to modify any of these plans as need be during the course of the class; however, I won't do anything too drastic, and you'll be informed as far in advance as possible.

Students are expected to follow the GT honor code as described here. Some points to keep in mind: Plagiarizing is defined by Webster's as to steal and pass off (the ideas or words of another) as one's own: use (another's production) without crediting the source. If caught plagiarizing, you will be dealt with according to the GT Academic Honor Code. Submitting any unattributed work other than your own is a violation of the Academic Honor Code. When in doubt, please ask the instructor.

I encourage you to discuss the assignments, ask questions about how to program, etc. with the instructor, TA, and other students, especially on Piazza. But the code you submit must be your own. Unauthorized copying of anyone else's code is a violation of the Academic Honor Code. Unauthorized reuse of code from online is a violation of the Academic Honor Code. You're taking this class to learn how to think about and create Game AI code on your own.

Learning about algorithms from text or online sources is permissible. Copying code verbatim from online or another student is not permissible.

I heavily encourage students to use art assets that are fair use (e.g. Creative Commons licensed audio, textures, images, etc.) or to produce their own. Unattributed use of other instancial assets, such as graphics, text, or audio, or use of such assets without reappropriating them in a meaningful way to make a clear unique contribution on the student's part is a violation of the Honor Code. Reuse of outside insubstantial assets is permissible, in other words, but must be done in a manner that makes it clear that you have made a major and significant contribution to the project in question. Failure to do so will result in a failing grade.

Failure to cite your sources is an Honor Code violation. Unauthorized use of any previous semester course materials, such as tests, quizzes, homework, projects, and any other coursework, is prohibited in this course. Using these materials will be considered a direct violation of academic policy and will be dealt with according to the GT Academic Honor Code.

All assignments are to be completed individually unless otherwise stated. For the assignments, do not copy from others or let others copy your work; however, you may think of the TAs as your close collaborators. Feel free to go to them with any problems that you are having. That's why they are there. You may speak with other students in the class about programming assignments. If you do so, I recommend that you do not do so in front of a computer. This will prevent the temptation to copy code. If you need to write something to communicate with a peer about an assignment, write by hand on a whiteboard or piece of paper. Erase the board (or

destroy the paper) and do not take a picture of the board. The whiteboard policy is designed to protect you: I cannot tell who copied whom and both may be persecuted.

Do not store your programming assignment solutions on an un-protected online source repository (e.g., Bitbucket, GitHub, etc.). This invites cheating with repercussions to both parties. It goes without saying that one should not download solution code if found on the Internet.