Class Assessments

Grade Categories

Your final grade in this class will be based on a number of components.

Category	% Weightage	Description
Assignments (A1-A6)	60	We will consider your best 5 scores achieved from the 6 assignments. Each assignment carries a 12% weightage. Whiteboard level collaboration is encouraged, but students should not look at working code.
Exams	35	There would be 2 week-long, open-book, open-notes, open-video lectures exams: Midterm (15%) and Final (15%). No collaboration of any kind is allowed.
Practice Quizzes	0	Class modules include simple example problems called "Practice Quizzes." The videos, readings, and the example problems for the topic should be completed before the week of the topic so that students are prepared for the weekly exercises.

Python Introductory Assignment (A0)	1	This is an assignment to help with the setup of the Python local environment.
Plagiarism Quiz	4	This is a quiz designed to inform about plagiarism policies in the class.
Extra Credit	Varies	Throughout the class, there will be opportunities to receive extra credit. These opportunities encourage a deeper, more “research-y” understanding of the material.

It is important to note that this course does *NOT* follow the normal grading buckets (90 or above for "A", 80 to 90 for "B", etc.). Make sure to pay attention to the announcements after each assignment is graded to understand where your grade sits in the big picture.

Achieving a final grade *above* the median will result in an “A”. A “B” will be given for final grades *equal* to the median and *above* 1 standard deviation below the median. Final grades *equal* to or *below* 1 standard deviation below the median and *above* 2 standard deviations below the median will get a “C”. Final Grades *equal* to or *below* 2 standard deviations below the median and *above* 3 standard deviations below the median will get a “D”. Any grade *equal* to or *below* 3 standard deviations below the median will get an “F”. There will be chances to earn extra credit points during the semester, which will be factored in at the end after all other curving is done, i.e. extra credit points will not contribute to the curves.

It's important to note that curving cannot give you a worse letter grade than the one you would receive under the normal grading buckets, i.e you can guarantee yourself a letter grade “A” by getting a grade *above* 90, and the same applies to other letters (e.g. 80-90 is “B”, etc.). Put another way, the cutoff for an "A" will be min(90, class_median), the cutoff for a "B" will be min(80, class_median - class_std_dev), and so on.

To compute your grade, first compute a fraction where the sum of the total number of possible points in any given category is the denominator and the total number of points you achieved in that category is the numerator. Then multiply that fraction by the weight of the category. Sum this computation across categories to retrieve your grade. Note that since each Assignment is worth the same number of points and we count 5 Assignments, each Assignment has a weight of 12%. Extra Credit will be weighted based on the category it is earned in. You can simply add it as "free" points to the total number of points you achieve in the category it is offered in. Finally, please note that even if an Assignment is dropped, the extra credit you earn on that Assignment is still counted towards your final grade. You can simply add these points to another Assignment since the Assignments are equally weighted.

Although we understand the importance of grades, we encourage you to focus first on doing the best you can; if you do, your grade should take care of itself.

Grading Policies

We will strive to return grades of the assignments and the midterm exam within two weeks of submission. Grades will generally be delivered via Canvas.

Note that grades on the last assignment and the final exam will be posted very close to the final grade submission deadline. Make sure to allocate time after finals to check your grades and make sure everything, especially these last two grades, are as you expect.

Lastly, remember that this class is effectively graded on a curve. If you try to interpret your grade according to the traditional categories, you will likely think you are doing worse in the class than you actually are. Make sure to pay attention to the stats posts at the end of each assignment for the context necessary to interpret your grade and evaluate your performance.

Remember that the cutoff for an A will be at most 90%. The cutoff for a B will be at most 80%. For a C, at most 70%, etc.

Assignments

There are six assignments in this class. Each assignment is open for about two weeks. In order to reduce the stress of illness or unavoidable conflicts, only the top five grades will be used in determining the final grade. However, we suggest you do not skip any and complete all of the assignments. The assignments will help with your understanding and your performance on the final exam. Each year, several students’ letter grades would have been higher had they completed the last assignment (which is one of your instructor’s favorite topic). Do not skip an assignment for convenience in the beginning of the semester, thinking that you can make it up on later assignments. This strategy almost always ends badly.

Make sure to attempt extra credit sections for each assignment that has it; extra credits will be added on top of your assignment score, even if you scored full marks already.

Note: the top five grades policy will not be applied in cases of academic violation. If it is your first academic conduct violation, the zero score grade will be enforced as one of the top five grades. Harsher penalties may be enforced for any academic conduct violations beyond the first one.

Most assignments will involve programming in Python. You may be wondering why we chose Python given that Peter Norvig and Thad Starner both prefer Lisp for teaching AI and that Alan Kay once called Lisp “The greatest single programming language ever designed”. In preparing for this course, the AI instructors surveyed and believed Python was the best compromise; it has inherited many good features of Lisp, is commonly used in industry (e.g., Google), and best matches the pseudocode in the book (according to Norvig himself). Students taking a course at this level should be able to become functional in a new language quickly. Please become acquainted with Python.

Below is a summary of the assignments. Due dates can be found on the **course calendar (https://gatech.instructure.com/courses/371008/pages/class-schedule)** .

1. Search (https://gatech.instructure.com/courses/371008/assignments/1588768) : Experiment with various search techniques to discover the most efficient way to find the shortest path between three places in a city.

2. Isolation Player (https://gatech.instructure.com/courses/371008/assignments/1588772) : Use the MINIMAX and alpha-beta pruning techniques and experiment with evaluation functions to create a program that can play a variant of the game Isolation better than a human.

3. ($CANVAS_OBJECT_REFERENCE$/assignments/gee03d094f4ba16b97d9dba086cac3c97) Bayes Nets (https://gatech.instructure.com/courses/371008/assignments/1588774) : Implement Bayesian networks and sampling algorithms to gain a better understanding of probabilistic systems.

4. Decision Trees (https://gatech.instructure.com/courses/371008/assignments/1588778) : Build, train, and test several decision tree models to perform basic classification tasks.

5. Gaussian Mixture Models (https://gatech.instructure.com/courses/371008/assignments/1588780) : Implement k-means clustering and Gaussian mixture models to perform basic image segmentation. Research, implement, and test the Bayesian Information Criterion to guarantee a more robust image segmentation.

6. ($CANVAS_OBJECT_REFERENCE$/assignments/gb99b1e4747362781d370ed10e9a313b6) Hidden Markov models (https://gatech.instructure.com/courses/371008/assignments/1588782) : Implement the Viterbi and Forward-Backward algorithm to recognize signals using HMMs.

*More information about the projects and their learning goals will be provided in the individual project assignment pages.

These python assignments are submitted to Gradescope (mandatory) and are auto-graded.

Midterm and Final

There will be a cumulative midterm and a final exam in this class. These exams will be take-home and will be open-books, open-notes, and open video lectures. However, the use of the internet is strictly prohibited, unless otherwise specified. The exams will be provided and submitted in PDF format. Gradescope will be used for grading, so students will be able to view their grades and examine the graded PDF for each exam.

It should be noted that these exams will have more application-based questions, which are designed to test your knowledge and require you to not only understand the material but also apply it correctly.

There is no collaboration of any kind on the midterm and final.

Weekly Exercises and Practice Quizzes

Often the videos, readings, and practice quizzes in the course modules fall by the wayside while focusing on the assignments and other aspects of life. Many students find it hard to maintain the discipline to complete the course modules every week without an incentive. Exercises help keep students on pace and are due at the end of the week the topic is covered.

Only N-2 of the N exercises will count for grades. This policy allows students to not have to ask for extensions when they become ill or something interferes with classwork.

For on-campus students, the exercises can be done collaboratively in class. On-line students can attend live remotely. For students who can not attend, they can follow along with the video recordings of class or do the exercises independently. Collaboration on these exercises is highly encouraged. Peer learning can be quite helpful for understanding AI concepts.