CS 4100: Artificial Intelligence (Spring 2025)

1 General Information

Time: Asynchronous | Location: Online

2 Teaching Staff

- **The preferred platform for asking questions and contacting staff is Piazza.**
  `https://piazza.com/northeastern/spring2025/cs4100online`
- If e-mail contact is necessary (e.g., sending attachments),
  the preferred e-mail address that reaches all staff is `cs4100-staff@ccs.neu.edu` .
- Only e-mail individual staff if absolutely necessary (e.g., confidential issue), and note that response will typically be slower than contacting all staff via Piazza or the staff mailing list.

Role	Name and E-mail	Office Hours	Location
Instructor	Lawson L.S. Wong `lsw@ccs.neu.edu`	Fri 2:30–4:30 PM and by appointment	Zoom 617 373 7459
TA	Satvika Eda `eda.s@northeastern.edu`	Tue 5–7 PM	Zoom 555 368 7443
TA	Shashank Manjunath `manjunath.sh@northeastern.edu`	Thu 4–6 PM	Zoom 807 670 6674
TA	Yu Qi `qi.yu2@northeastern.edu`	Wed 12–2 PM	Zoom 757 826 7655
TA	Jiayi Zhang `zhang.jiayi12@northeastern.edu`	Mon 4–6 PM	Zoom 201 245 7553

3 Course Overview

This course will introduce students to the fundamentals of artificial intelligence, including the broad areas of search, decision-making under uncertainty, machine learning, and probabilistic inference.

The above topics only cover a small portion of the entirety of AI, and do not even cover all of the fundamentals (prominent topics that will not be covered include logical reasoning and knowledge representation). The emphasis is on well-established foundational methods and frameworks, instead of the cutting edge and latest developments (e.g., large language models). However, by the end of the course, students will have developed a sufficiently broad set of technical tools, that will enable them to solve many real-world problems, self-learn additional techniques, understand advances in the field more deeply, and pursue further specialized courses in AI.

The course material will focus on problem types, models, and algorithms. Applications will be discussed when relevant, but will not be the focus of the content. However, in the spirit of experiential learning, there will be significant opportunities for implementation and application, through the programming assignments and the project.

4 Textbook and Reference Materials

There is no required textbook. However, the following materials are recommended:

- The standard AI textbook is *Artificial Intelligence: A Modern Approach* (AIMA; 4th edition), by Stuart Russell and Peter Norvig. This textbook serves as excellent reference material, and this course builds on the book's content. If you are considering pursuing further studies in AI, obtaining and reading this textbook is highly recommended. The 3rd edition is reasonably similar to the current edition (with different chapter/section numbers).
  `http://aima.cs.berkeley.edu/index.html`

- This course is heavily influenced by the CS 188 course at UC Berkeley,
  developed by Dan Klein and Pieter Abbeel and many others.
  - The course archives offer additional lecture slides, videos, and practice problems:
    `https://ai.berkeley.edu`
  - Lecture videos from CS 188 Spring 2024:
    `https://www.youtube.com/playlist?list=PLp8QV47qJEg67UTShQ4er4RYQ3rOeDKxv`

5 Prerequisites

- All programming assignments must be completed in Python 3.
- Later in the course you will need to use basic probability. A short refresher will be provided, but it would help to learn this as soon as possible.
- Even later in the course you will need to use basic single-variable differential calculus. A short refresher will be provided.

6 What is Where

We will use Canvas to host course materials and use Piazza as our primary communications hub.

- Canvas: `https://northeastern.instructure.com/courses/202258`
  Content: Links to lecture videos, lecture slides/notes, assignments, grades, other materials. All official course-related content will be posted on Canvas. All assignments should be submitted on Canvas, and all grades/feedback will be posted on Canvas.

- Piazza: `https://piazza.com/northeastern/spring2025/cs4100online`
  Content: Announcements, Q&A, discussion.
  Preferred platform for contacting course staff.
  All communications-related activities will be conducted on Piazza, including official course announcements. The site also offers an excellent discussion forum, where both instructors and fellow students can answer questions. Everyone is encouraged to participate. Questions/notes can be posted anonymously or with identity, and may also be posted privately only to instructors. Note that posting questions/notes via Piazza will most likely result in faster responses compared to e-mailing individual instructors.

7 Coursework

Type	Frequency	Due dates	Evaluation
Exercises	$\sim$ Biweekly (4 total)	Friday (11:59 PM ET)	25%
Programming assignments	$\sim$ Biweekly (4 total)	Friday (11:59 PM ET)	25%
Exams	2 total	Approximately 2/21, 3/31	20%
Project	1 total	See schedule below	30%

- Exercises are based on the previous two weeks of material. Students may discuss the problems with other students, but must write up their own solutions. On each assignment, please also indicate who you discussed with (if any). Also see Collaboration Policy below.
  Lateness: Up to two days late (24-hour period), penalized by 5% per day.

- Programming assignments are designed to let you see algorithms working in practice. Students should work on this by themselves. Resist the temptation to search for existing solutions – the process of implementation and debugging is critical to learning the material.
  Lateness: Up to two days late (24-hour period), penalized by 5% per day.

- There will be 2 midterm exams, administered asynchronously online, during the weeks of 2/17–2/21 and 3/31–4/4. The specific logistics of exam administration will be announced early February. They will cover the first and the {second + first half of third} modules of the course respectively, and are not cumulative. See the course schedule for details.
  There is no final exam (but there is a final project).

- The project offers an opportunity to apply learned techniques on a substantial problem that interests the student. Further details, (non-exhaustive) topic suggestions, and examples of projects from previous semesters will be provided in February.
  Here is a rough timeline for the project, but is subject to change:

  - February 21: Project pre-proposal
  - $\sim$ March 11: Project proposal
  - April 11: Milestone
  - April 18: Draft report
  - April 24: Final report

8 Academic Integrity and Collaboration Policy

We encourage collaboration and discussion in the class, as long as help is fully acknowledged. Discussion of high-level ideas is generally permitted (except during exams), but submissions should always be your own work (except project, which should be your team's work). For specific collaboration constraints on different parts of the course, see the Coursework section above.

This collaboration policy applies equally to recent AI tools such as ChatGPT and other large language models (LLMs). You are welcome to explore these tools and engage in discussion with them, but you should not be asking them to solve assignments, and you should not be copying answers/code from them. Additionally, if you use these tools, you should acknowledge usage and provide details about how you are using them.

Cheating and other acts of academic dishonesty will be referred to OSCCR (office of student conduct and conflict resolution) and the Khoury College of Computer Sciences.

9 Schedule (subject to change; version 20250117)

Week	Lec #	Topic	Reference (AIMA 4e)	Assignments due (Fri 11:59 PM ET)
Jan	1	Course overview	1–2	
6–10	2	Uninformed search	3.1–3.4	
Jan	3	Uninformed search (continued)	3.4	
13–17	4	Heuristic search	3.5	
				PA 1 (Q1–Q3)
Jan		*Martin Luther King, Jr. Day*		
20–24	5	Heuristic search (continued)	3.5–3.6	
				Ex 1
Jan	6	Adversarial search	5.1–5.2	
27–31	7	Adversarial search (continued)	5.3–5.5	
				PA 1 (all parts)
Feb	8	Decision theory	16.1–16.3	
3–7	9	Markov decision processes (MDPs)	17.1–17.2	
				Ex 2
Feb	10	Bellman equation	17.1–17.2	
10–14	11	Value iteration	17.2	
				PA 2
Feb		*Presidents' Day*		
17–21		**Exam 1**	Lectures 2–7	
				Project pre-proposal
Feb	12	Reinforcement learning	17.3, 22.1–22.2	
24–28	13	Q-learning	22.2–22.4, 22.7.1	
				Ex 3
Mar		*Spring break*		
3–7				
Mar	14	Machine learning	19.1–19.2	*Project proposal*
10–14	15	Linear regression	19.6–19.6.2	
				PA 3
Mar	16	Logistic regression	19.6.4–19.6.5, 20.2.1	
17–21	17	The practice of machine learning	19.4, 19.9	
				Ex 4
Mar	18	Neural networks and deep learning	21.1–21.2, 21.4	
24–28	19	Bayesian inference	12.3–12.5	
				PA 4
Mar 31		**Exam 2**	Lectures 8–17	
– Apr 4	20	Hidden Markov models (HMMs)	14.1–14.3, 14.5.3	
Apr	21	Partially observable MDPs	17.4–17.5	
7–11	22	Topics: Frontiers (TBD)		
				Project milestone
Apr	23	Topics: Research (TBD)		
14–18				*Draft report due 4/18*
Apr		*Patriots' Day*		
21–25				*Final report due 4/24*

10 Learning Objectives

Module 1: Search – Sequential decision-making under certainty

- Concepts: Agents, environments, states, actions, graph search, tree search, heuristics
  Properties of search algorithms: Time/space complexity, optimality, soundness, completeness
  Further types of search problems: Adversarial search (game tree)
- Algorithms: BFS, DFS, IDS, UCS, Greedy search, A* search
  Understand how the above algorithms are all unified by a priority queue
  Further algorithms: Minimax search, expectimax search
- Mastery objective: Given a sequential decision-making problem (with deterministic outcomes), formulate it as a search problem (by specifying formal components of a search problem), and solve the problem with an appropriate algorithm and heuristics (if applicable).

Module 2: MDPs – Sequential decision-making under uncertainty

- Concepts: Probability (expectation, conditional), utility, maximum expected utility
  Markov decision process (MDP): Reward, return, value, policy, Bellman equation
  Reinforcement learning (RL): Exploration vs. exploitation, model-based vs. model-free, Q-function (action-value function), temporal-difference error, linear function approximation
- Algorithms: Value back-up (and-or tree), value iteration (dynamic programming), Q-learning
- Mastery objective: Given a sequential decision-making problem (with stochastic outcomes), formulate it as an MDP (by specifying formal components of an MDP), and solve the problem with an appropriate MDP/RL algorithm.

Module 3: Learning – Acquisition of knowledge and inference

- Concepts: Categories of machine learning, supervised learning, regression, classification
  Learning problem: Dataset, hypothesis, parameters, loss/error function, learning algorithm
  Further concepts: Maximum likelihood, hyperparameters, model selection, train/validate/test
  Inference: Probability (conditional, marginalization), Bayes' rule, Bayesian inference
  Graphical models: Hidden Markov models (HMM), Bayesian networks, generative models
- Models: Linear regression, logistic regression, (deep) neural networks
  Algorithms: Coordinate descent, (stochastic) gradient descent, cross validation
- Mastery objective: Given a learning problem with an optimization objective, derive an appropriate learning algorithm, and understand how to apply the algorithm in practice.

11 Grades

The final grade cutoffs change semester to semester due to variations in the assigned work and evaluation metrics. Based on past experience, the following ranges should be expected:

- 85–100: A range (A- or A)
- 65–85: B range (B- / B / B+)
- < 65: C range or below