# Spring 2016 Syllabus

This syllabus is subject to change. Note that unreleased project out and due dates are just guesses and will likely change somewhat.

| Day | Topic | Optional Reading | Slides | Video | Assignment | Due |
|---|---|---|---|---|---|---|
| Tu 1/19 | Introduction to AI | Ch. 1 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | Math Self-Diagnostic P0: Tutorial | (ungraded) F 1/22 5pm |
| Th 1/21 | Agents and Search | Ch. 3.1-4 (2e: Ch. 3) | 1PP · 2PP 4PP · 6PP PPT | Live Edited | HW1 (section 1) | M 2/1 11:59pm |
| Tu 1/26 | A* Search and Heuristics | Ch. 3.5-6 (2e: Ch. 4.1-2) | 1PP · 2PP 4PP · 6PP PPT | Live Edited | P1: Search Contest 1: Search | F 2/5 5pm Su 2/7 11:59pm |
| Th 1/28 | Constraint Satisfaction Problems | Ch. 6.1 (2e: Ch. 5.1) | 1PP · 2PP 4PP · 6PP PPT | Live Edited | HW2 (section 2) | M 2/8 11:59pm |
| Tu 2/2 | CSPs II | Ch. 6.2-5 (2e: Ch. 5.2-4) | 1PP · 2PP 4PP · 6PP PPT | Live Edited | | |
| Th 2/4 | Game Trees: Minimax | Ch. 5.2-5 (2e: Ch. 6.2-5) | 1PP · 2PP 4PP · 6PP PPT | Live Edited | HW3 (section 3/ exam-prep 1) | Tu 2/16 11:59pm |
| Tu 2/9 | Game Trees: Expectimax; Utilities | Ch. 5.2-5 (2e: Ch. 6.2-5) | 1PP · 2PP 4PP · 6PP PPT | Live Edited | P2: Multi-Agent Search Contest 2: Multi-Agent Search | F 2/19 5pm Su 2/21 11:59pm |
| Th 2/11 | Markov Decision Processes | Ch. 16.1-3 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | HW4 (section 4 / exam-prep 2) | Tu 2/23 11:59pm |
| Tu 2/16 | Markov Decision Processes II | Sutton and Barto Ch. 3-4 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | | |
| Th 2/18 | Reinforcement Learning | Ch. 17.1-3, S&B Ch. 6.1,2,5 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | HW5 (section 5 / exam-prep 3) | M 2/29 11:59pm |
| Tu 2/23 | Reinforcement Learning II | | 1PP · 2PP 4PP · 6PP PPT | Live Edited | P3: Reinforcement Learning | F 3/4 5pm |
| Th 2/25 | Probability | Ch. 13.1-5 (2e: Ch. 13.1-6) | 1PP · 2PP 4PP · 6PP PPT | Live Edited | HW6 (section 6 / exam-prep 4) | M 3/7 11:59pm |
| Tu 3/1 | Bayes' Nets: Representation | Ch. 14.1-2,4 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | | |
| Th 3/3 | Bayes' Nets: Inference | Ch. 14.3, Jordan 2.1 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | (section 7 / exam-prep 5) | |
| Tu 3/8 | Bayes' Nets: Sampling | Ch. 14.4-5 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | | |
| Th 3/10 | Decision Networks / VPI | Ch. 15.1-3, 6 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | Practice Midterm / Solutions | Sa 3/12 11:59pm |
| Tu 3/15 | MIDTERM (7-10p) | | | | | |
| Th 3/17 | VPI and Markov Models | Ch. 15.2-5 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | P4: Bayes Nets | M 4/4 11:59pm |
| Tu 3/22 | Spring Break | | | | | |
| Th 3/24 | Spring Break | | | | (section 8 / exam-prep 6) | |
| Tu 3/29 | HMMs | Ch. 15.2,6 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | HW7 | Tu 4/5 11:59pm |
| Th 3/31 | HMM Filtering | Ch. 15.2,6 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | (section 9 / exam-prep 7) | |
| Tu 4/5 | ML: Naive Bayes | Ch. 15.2,6 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | P5: Ghostbusters HW8 | F 4/15 5pm Tu 4/12 11:59pm |
| Th 4/7 | ML: Perceptrons | Ch. 15.2,6 | 1PP · 2PP 4PP · 6PP PPT | Live Edited | Final Contest (section 10 / exam-prep 8) | Su 4/24 11:59pm |
| Tu 4/12 | ML: Deep Learning I | | 1PP · 2PP 4PP · 6PP PPT | Live | | |
| Th 4/14 | ML: Deep Learning II | | 1PP · 2PP 4PP · 6PP PPT | Live | HW9 (section 11 / exam-prep 9) | Th 4/21 11:59pm |
| Tu 4/19 | Advanced Topics: Robotics | | 1PP · 2PP 4PP · 6PP PPT | Live | | |
| Th 4/21 | Advanced Topics: Natural Language Processing | | 1PP · 2PP 4PP · 6PP PPT | Live | HW10 (section / exam-prep) | F 4/29, 5pm |
| Tu 4/26 | Advanced Topics: Human Robot Interaction | | 1PP · 2PP 4PP · 6PP PPT | Live | P6: Classification | TBD |
| Th 4/28 | Advanced Topics / Final Contest | | 1PP · 2PP 4PP · 6PP PPT | Live | Practice Final | Sa 5/7 11:59pm |
| Th 5/12 | FINAL EXAM (8:00-11:00am) | | | | | |

# Course Policies

## TIMES AND LOCATIONS

- **Lecture:** Tuesdays and Thursdays 11:00-12:30pm in 155 Dwinelle
- **Discussion Sections:** Beginning the week of Jan 25.
  - 102: Mon 10-11am, 102 Latimer (Christopher)
  - 104: Mon 11-12pm, 102 Latimer (Christopher)
  - 106: Mon 12- 1pm, 229 Dwinelle (Davis)
  - 107: Mon 1- 2pm, 283 Dwinelle (Davis)
  - 109: Mon 2- 3pm, 105 Dwinelle (Karthik)
  - 111: Mon 3- 4pm, 258 Dwinelle (Karthik)
  - 113: Mon 4- 5pm, 205 Dwinelle (Greg)
  - 116: Mon 2:30-3:30pm, 310 Soda (Jacob)
  - 117: Mon 3:30-4:30pm, 310 Soda (Jacob)
  - 118: Mon 6-7pm, 320 Soda (Greg)
- **Discussion Overflow Sections:** ONLY for the weeks of 1/25-1/29, and 2/1-2/5.
  - 101: Mon 9-10am, 289 Cory (Coline)
  - 103: Mon 10-11am, 105 Latimer (Tianhao)
  - 105: Mon 11-12pm, 310 Hearst Mining (Abhishek)
  - 108: Mon 1- 2pm, 205 Dwinelle (Abhishek)
  - 110: Mon 2- 3pm, 183 Dwinelle (Weicheng)
  - 112: Mon 3- 4pm, 228 Dwinelle (Weicheng)
  - 114: Mon 4- 5pm, 105 Dwinelle (Coline)
  - 115: Mon noon-1pm, 310 Soda (Tianhao)
- **Exam Practice Sessions:** Beginning the week of Feb 8.
  - 101: Mon 9-10am, 289 Cory (Coline)
  - 103: Mon 10-11am, 105 Latimer (Tianhao)
  - 105: Mon 11-12pm, 310 Hearst Mining (Abhishek)
  - 108: Mon 1- 2pm, 205 Dwinelle (Abhishek)
  - 110: Mon 2- 3pm, 183 Dwinelle (Weicheng)
  - 112: Mon 3- 4pm, 228 Dwinelle (Weicheng)
  - 114: Mon 4- 5pm, 105 Dwinelle (Coline)
  - 115: Mon noon-1pm, 310 Soda (Tianhao)

## COMMUNICATION

There will be several routes of communication for this course:

- Announcements will be posted to this website.
- The main mode of electronic communication between students and staff, as well as amongst students, will be through Piazza. It is intended for general questions about the course, clarifications about assignments, student questions to each other, discussions about material, and so on. We strongly encourage students to participate in discussion, ask, and answer questions through this site. The course staff will monitor discussions closely.

- If you need to contact the course staff privately, you should email cs188-staff AT lists.berkeley.edu. You may of course contact the professor or GSIs directly, but the staff list will produce the fastest response.

## COURSE DESCRIPTION

This course will introduce the basic ideas and techniques underlying the design of intelligent computer systems. A specific emphasis will be on the statistical and decision-theoretic modeling paradigm.

By the end of this course, you will have built autonomous agents that efficiently make decisions in fully informed, partially observable and adversarial settings. Your agents will draw inferences in uncertain environments and optimize actions for arbitrary reward structures. Your machine learning algorithms will classify handwritten digits and photographs. The techniques you learn in this course apply to a wide variety of artificial intelligence problems and will serve as the foundation for further study in any application area you choose to pursue.

See the syllabus for slides, deadlines, and the lecture schedule.

## PREREQUISITES

- **CS 61A or 61B:** Prior computer programming experience is expected (see below); most students will have taken both these courses.
- **CS 70 or Math 55:** Facility with basic concepts of propositional logic and probability are expected (see below); CS 70 is the better choice for this course.

This course has substantial elements of both programming and mathematics, because these elements are central to modern AI. You should be prepared to review basic probability on your own if it is not fresh in your head. You should also be very comfortable programming on the level of CS 61B even though it is not strictly required.

**CS61A AND CS61B AND CS70 is the recommended background.**

## LANGUAGE

Course programming assignments will be in Python. We do not assume that students have previous experience with the language, but we do expect you to learn the basics very rapidly. Project 0 is designed to teach you the basics of Python, but if you want to get a head start here is a good tutorial: ACM Python Tutorial

## ASSIGNMENTS

This class includes **six programming projects** and **regular electronic assignments**.

**Collaboration:** Project 0 is to be completed alone.  Projects 1 through 6 can be completed alone or in teams of two.  When completing in a team of two it is important that both team members submit their team's project files into edX to ensure a grade is registered for both team members. Homework is to be submitted individually but may be discussed in groups.

**Slip days:** Programming projects must be turned in electronically by 5:00pm on the listed due date. You will have a total of **5 slip days** for these projects, up to **two of which can be used for each project**. Note that slip days are counted at the granularity of days, rounded up to the nearest day. For example, for a project due at 5pm on Friday, any submission from Friday 5pm - Saturday 5pm will use up one slip day, any submission from Saturday 5pm - Sunday 5pm will use up two slip days, and any submissions after Sunday 5pm will receive no credit. The other homework assignments do not have late days!

**Project grading:** Projects will by default be graded automatically for correctness, though we will review projects individually as necessary to ensure that they receive the credit they deserve. To calculate your total project grade, we will consider all possible allocations of slip days across all projects to give you the best score achieved. For example, suppose you submitted a 24/25 on Project 1 on time, and then resubmitted a 25/25 on Project 1 after using two slip days. If by Project 5, you used up all your slip days, but you need to use an additional slip day on Project 5 and wish you had not submitted two days late for the one point in Project 1, you can submit into the late module for Project 5 (edX does not block you from going over the number of slip days) and we will automatically redistribute your slip days usage.

**Ethics:** Submissions should acknowledge all collaborators and sources consulted. All code and written responses should be original. We trust you all to submit your own work, but to protect the integrity of the course from anyone who doesn't want to play by the rules, we will actively be checking for code plagiarism (both from current classmates and previous semesters). I'm not lenient about cheating; I sympathize with Kris Pister's policy.

## GRADING

Overall grades will be determined from:

- Programming Assignments (30%)
- Electronic Assignments (10%)
- Midterm Exam (25%)
- Final Exam (35%)

Grades are on the following fixed scale:

|    |    |
| --- | --- |
| A | [90 – 100]% |
| A- | [85 – 90]% |
| B+ | [80 – 85]% |
| B | [75 – 80]% |
| B- | [70 – 75]% |
| C+ | [65 – 70]% |
| C | [60 – 65]% |
| C- | [55 – 60]% |
| D+ | [50 – 55]% |
| D | [45 – 50]% |
| D- | [40 – 45]% |
| F | [0 – 40)% |

These cutoffs represent grade minimums. We may adjust grades upward based on class participation, extra credit, etc. The grade of A+ will be awarded at the professor's discretion based on exceptional performance.

**Regrade Policy:** If you believe an error has been made in the grading of one of your exams or assignments, you may resubmit it for a regrade. Regrades for cases where we mis-applied a rubric in an individual case are much more likely to be successful than regrades that argue about relative point values within the rubric, as the rubric is applied to the entire class. Because we will examine your entire submission in detail, your grade can go up or down as a result of a regrade request.

## TEXTBOOK

No textbook is required for this course. For students who want to read more, we recommend Artificial Intelligence: A Modern Approach by Stuart Russell (UC Berkeley) and Peter Norvig (Google). Be aware, however, that this is not a course textbook, so our presentation does not necessarily follow the presentation in the book.

## ENROLLMENT

Here are the policies that govern admission into classes, and here are some answers to frequently asked questions about admission. The course staff does not control enrollment!