Overview

By the end of this class, you will know when and how to deploy the fundamental techniques for building intelligent systems. We’ll cover concepts and algorithms in areas such as combinatorial search and decision making, knowledge representation and reasoning, planning, reasoning under uncertainty, and learning. You will gain experience implementing these concepts and algorithms as complete working programs. You should already be a fluent programmer, understand common data structures, and be familiar with basic complexity analysis and big-O notation. This class counts as ‘implementation-intensive’ in the UNH curriculum.

Contact Info

Prof. Wheeler Ruml, ruml at cs.unh.edu, Kingsbury N215D, 2-2683
Office hours: Mondays 2:30-3:30pm

TA Bence Cserna, bence at cserna.net, Kingsbury W326
Office hours: Wednesdays 11:30am-12:30pm

Please feel free to drop by office hours for any reason, even just to chat about AI. If you can’t make office hours, contact us to set up a time to meet.

If you have questions that you’d like answered quickly, post on the forum at piazza.com. If you email either of us directly with a question regarding an assignment, we will probably just ask you to post on piazza (this way, everyone can see the response). I recommend setting piazza to send you email for new posts — important announcements are sometimes made!

Required Text


Evaluation

These breakdowns are tentative and can change at any time.
Non-writing intensive:

70% 12 programming assignments.
30% final project, including a proposal, oral presentation, and final paper.

Writing intensive:

65% 12 programming assignments.
30% final project, including a proposal, oral presentation, and final paper. Extra weight on your paper, which you must submit a draft of before the deadline for feedback.
5% entries on the class blog, posted weekly, of your thoughts on the course content. This can include, for example, outlines of potential class projects, ideas for commercial applications, and discussions of algorithms.

Please note that assignment deadlines are not flexible and there will be no credit for late work. This allows us to discuss solutions promptly after the deadline. The schedule indicates when assignments are due so that you can plan your work in advance.
Please don’t come to class if you are going to sleep, eat, or use a phone or laptop. Your physical presence is required only for the exams and the project presentation. However, useful stuff is discussed in class, so I recommend attending!

If you find emotional or mental health issues affecting your performance, please contact the UNH counseling center (www.unh.edu/counseling-center) — they are often very helpful!

Except for the final project, which may be done in a team, all your work in this class should be your own. Any collaboration must be cleared with me in advance and you must cite all sources you use in preparing your work, other than the textbook, lectures, and recitations. If you talk to others in depth about the concepts for an assignment, you should list them on your assignment. Do not talk about your assignment code with others. As a scientist, I take a dim view of academic dishonesty and you will fail the class and be reported to the Dean for possible dismissal from the University if you cheat.

If you are registered with the student disability office, please let me know as soon as possible so that I can provide proper accommodation.

Mechanics

Programming assignments:

You may use any programming language you wish in this course. You will get example inputs for your programs that indicate the sort of problems your solutions will be tested on. You code must run on agate.cs.unh.edu. We sometimes supply some test inputs, but these are not exhaustive — you should find and test edges cases yourself. See the submission information on the web page for make script requirements and instructions on submitting. You will use our submit script for both your source code and your answers to the written questions for each assignment. Please hand in a hardcopy of the written answers to the TA in class, but don’t bother printing your source code. Please include in your code submission a transcript of your code running that provides evidence for each aspect that works. We will also ask you to clearly explain in your written answers which aspects of your code do not work.

It is in your best interest that your code be clear and concise, so that we can quickly understand it if we are trying to give you partial credit. You want your grader to be happy! Please bring any questions regarding grades or your work to our attention as soon as possible. You may ask that any of your work be regraded up to 7 days after it is handed back, but the new grade will replace the old one whether it is higher or lower.

Final project:

See the separate sheet for information about the final project, presentation, and paper.

Weekly Writing (writing intensive only):

Those taking the writing intensive version of the class must post an entry of at least 400 words to the course blog before each Tuesday’s class. I encourage three kinds of posts: 1) design ideas for your final project, 2) critiques and extensions of ideas from class or the textbook, and 3) ideas for applications of the ideas from class. Other kinds of posts are okay too, but your post must go beyond a superficial reaction to the course material (“I thought X was so cool! It reminds me of this other crazy thing...’) and actually engage in critiquing or extending the ideas (“Algorithm X can’t handle problems like Y because of Z, but if we did Q, then it might work because...”). If you are unsure if a particular topic is appropriate for a blog entry, feel free to ask.

Anyone in the class is welcome to post blog entries or comments on others’ entries. Let the course staff know if you don’t have access to the blog.