Csc416 Course Syllabus, Fall 2022

Csc416 is the the first course in a sequence of two courses in **symbolic artificial intelligence** at SUNY Oswego.

Instructor - Class Meetings - Office Hours - Email Processing Hours

Classes for the Fall 2022 edition of Csc416 will be offered in **face-to-face / in-person mode**, rather than any sort of remote or hybrid mode, so long as the college is not required to go remote, once again, due to the pandemic or some other unforeseen circumstance.

- Instructor: Craig Graci, Computer Science & Cognitive Science
- Class meetings: Monday, Wednesday & Friday 11:30am-12:25pm - **face-to-face / in-person**
- Office Hours: Monday 7:30am-8:30am, Tuesday 9:00am-10:00am, Friday 7:30am-8:30am - **Google Meet, by appointment only**
- Email Processing Hours: Monday 2:00pm-3:00pm & Friday 2:00pm-3:00pm

Text

The **suggested** texts represent, in an abstract way, the content of the course. The **required** text is meant to provide you with a much broader perspective of the field that I plan to share with you in lectures, and to position you to chat with the person in the street or on the elevator who might like to engage you in talk about AI should they learn that you have taken a course in the field.

- **Suggested:** A book on AI that features knowledge representation and search with applications to a number of cognitive topics (e.g., problem solving, planning, reasoning, game playing).
- **Suggested:** A book on Common Lisp.
- **Required:** Mitchell, M. *Artificial Intelligence: A Thinking Guide for Humans*.

Course Description

This course is essentially a computer programming based introduction to artificial intelligence. Emphasis on programming encourages depth of understanding, albeit at the expense of coverage. In order to enhance the scope of coverage and to provide a balanced perspective, reading/study of the required text will be incorporated into the course. The programming thread of the course will run more or less independently of the reading thread. That said, there will be some overlap between the two threads, and a concerted effort will be made to observe connections between the two.

With respect to the principle thread, the programming thread, elements of Common Lisp will be introduced by means of lectures, examples, and a number of programming challenges. These lectures, examples, and programming challenges will feature a selection of basic tools, techniques, and methodologies associated with the field of artificial intelligence, including state space representation, game playing, heuristic search, rule-based systems, and genetic algorithms. We will commence our study of Lisp with elementary data types and essential forms, and culminate our study with an in depth examination of CLOS, the Common Lisp Object System. In between we will, among other

things, learn about association lists and property lists, perform recursive list processing, and program with higher order functions. In short, we will explore a host of programming language concepts and constructs. As the semester proceeds, we will hone our Lisp skills by applying our growing knowledge of the language to a range of activities that pertain to the development of AI systems.

A third thread running through the course will be the maintenance of a personal online course portfolio. The nature of this portfolio will be developmental. It is intended that the portfolio store artifacts that result from course related programming and problem solving activities, and reference interesting resources that pertain to AI. These resources should include Lisp documentation and tutorials, as well as sites of various sorts that illuminate phenomena associated with the field of AI.

Learning Outcomes

Upon successful completion of this course it is expected that you will be able to:

1. Effectively engage in conversation about the nature and scope of artificial intelligence.
2. Discuss historical events associated with the field of artificial intelligence.
3. Write simple Lisp programs that feature association lists, property lists, and various other sorts of symbolic expressions.
4. Write Lisp programs that feature the writing of recursive functions and the application of higher order functions.
5. Perform Lisp programming with classes and objects in CLOS.
6. Describe standard knowledge representations that fall within the realm of symbolic artificial intelligence.
7. Write programs to explore aspects of search within symbolic AI systems.
8. Design a rule-based system.
9. Craft a state space problem solver.
10. Engage in heuristic programming.
11. Code a genetic algorithm.
12. Articulate ideas and issues associated with the raging debates pertaining to the promise and the perils of AI.

Teaching Model

In order to effectively learn something, it is helpful to rely on a suitable model of learning to guide your progress. In college, your professors generally establish some constraints on your engagement with course material which will, in turn, constrain whatever model of learning you might establish for yourself. For the present course, these are the principle constraints of engagement:

1. **Classroom Presence**: Come to class prepared (1) to share material pertaining to the various learning activities in which you are expected to engage, and (2) to contribute something meaningful when asked for a contribution of some sort, perhaps an idea pertaining Lisp programming, or a thought on how to solve a problem, or a suggestion for an appropriate knowledge representation with respect to a particular AI related task.
2. **Programming Challenges**: You will be asked to engage in a number of programming activities. Some will call on you to mimic an interactive session that I will share with you, perhaps with a bit of modification. Some will ask you to perform specific programming tasks in Lisp or in CLOS. Others will call on you to build components of a planner or a problem solver or another sort of AI program, and perhaps explore the behavior of the system in one way or another.

3. **Problem Sets**: You will be asked to do a small number of problem sets which are computationally oriented but do not call upon you to write or demo computer programs.
4. **Web Site**: You are required to build a web site, subject to a number of constraints, on which to place your work for this course. The web site will provide you with an opportunity to demonstrate that you can craft an admirable on-line portfolio of work. By taking seriously the task of presenting solutions your assignments in a clear manner for all to see, you are likely to improve your skills with respect to artificial intelligence programming, and building of components of AI systems. I will occasionally be calling upon one or another of you to present work by walking us through some of the items that you are expected to archive on your work site.
5. **AI Reading/Mining/Discussion Tasks**: You will be asked to spend some time with a number of chapters in Melanie Mitchell's text on AI, and for each chapter, (1) read the chapter, (2) answer a few questions on the chapter, (3) post your answers to your web site, and (4) engage in discussion of the chapter when class time is allocated for that purpose.
6. **Exam 1**: There will be a 1 hour closed-book exam during one of the class periods within week 6 or week 7 of the semester. The exam will be formally announced one week prior to the exam date.
7. **Exam 2**: There will be a 1 hour closed-book exam during one of the class periods within week 12 or week 13 of the semester. The exam will be formally announced one week prior to the exam date.
8. **Final Exam**: There will be a 2 hour exam during the officially scheduled final exam period, which is Wednesday, December 7, from 10:30am to 12:30pm.

My role as teacher will be to (1) orchestrate these learning activities, and (2) choreograph classes in a manner that integrates various aspects of these learning activities with the introduction and elaboration of material which is of central artificial intelligence concern.

Suggested Process of Engagement for Students

Considering the design of this course, the following informal procedure describes how I would engage in it, were I taking the course.

For each day of the semester:

- If it is a class day, I would be certain to attend class. Furthermore, if I am aware that a particular activity will be on the class agenda for the day, I would prepare to participate appropriately in the activity.
- I would scan my to do list for assignments and/or classroom exercises that are outstanding, and I would I would work in an appropriate manner towards completing them in a timely fashion. Specifically:
  - If the due date of an assignment is fast approaching (if the assignment is due in the next day or two), I would complete it, and post my work on the assignment to my web work site in the prescribed manner.
  - If an exercise was assigned during the previous class period, I would spend some time working on the exercise, endeavoring to reach a satisfactory state of completion on the exercise prior to the next class period.

Furthermore:

- When the first exam date is announced, and study materials are presented, I would work steadily, with intensity, to prepare for the exam. And, naturally, when the day of the first exam arrives, I would do my best to arrive at the exam in a relaxed, well-prepared state.
- When the second exam date is announced, and study materials are presented, I would work steadily, with intensity, to prepare for the exam. And, naturally, when the day of the second exam arrives, I would do my best to arrive at the exam in a relaxed, well-prepared state.
- When the time comes for you to perform the demo, I would make sure that I am well prepared to do so!

- When the last week of class rolls around, and study materials are presented, I would commence serious study for the final exam. And, naturally, when the day of the final exam arrives, I would do my best to arrive at the exam in a relaxed, well-prepared state.

Requirements

You are required to regularly attend class.

You are required to participate appropriately in classroom activities, including informal "discussions" and occasional student presentations (generally based on selected elements from the various programming assignments, problem sets, and exams).

You are required to complete all of the programming assignments.

You are required to complete all of the problem sets.

You are required to build a web work site on which to present your work on the programming assignments and problem sets.

You are required to fully participate in the reading/mining/discussion tasks associated with Melanie Mitchell's text on AI.

You are required to do a late semester demo of a number of the programs that you were assigned to write throughout the semester.

You are required to take two 1-hour regular season exams and a 2-hour final exam. These will be in-class exams. Furthermore, no cognitive artifacts, beyond blank paper and a writing stick, will be permitted for use on these exams.

Grading

Your grade will be determined on the basis of your performance on the following nonexclusive activities:

1. The class attendance (10 percent)
2. The web work site (10 percent)
3. Reading/mining/discussion tasks (10 percent)
4. The first exam (15 percent)
5. The second exam (15 percent)
6. Late semester demo (15 percent)
7. The final exam (25 percent)

Furthermore, I will adhere to the typical process for allocating grades. Thus, with respect to overall percentages, 90 or above will map to A, 80s will map to B, 70s will map to C, 60s will map to D, and other numbers will map to E.

On the Programming Assignments and Program Related Assignments

The programming assignments, and the programming related assignments, afford you learning opportunities that have the potential to position you to do well on the exams. Although you do not directly recived points for completing the programming assignments and the programming related assignments, you indirectly receive points by virtue of the direct relationship of assignment related knowledge to the composition of the exams. Generally speaking, if you

authentically obtain ownership of the knowledge related to the programming assignments, your efforts will likely be rewarded by enhanced performance on the exams.

What does it mean to authentically engage in doing an assignment? To authentically engage in a programming assignment means to do your best to do it on your own, using your problem solving skills and your knowledge of the programming language at hand to accomplish what needs to be done. If you run into a dead-end, will want to rely on "the three Rs of learning" to find a way forward. The three Rs of learning? Reflection, resourcefulness, and resilience. With respect to resourcefulness, you might look for a clue online, either at the work posted by other students in the class, or at related programs that you find on the web. Caution: Strive to take no more than the least amount of information that you need from these resources in order to continue working on your own. The more information that you take, the more you compromise the authenticity of your engagement with the assignment! As a rule, you should be mindful of the fact that there is a huge difference with respect to your relationship to knowledge obtained (1) by authentically construction a solution to a programming problem or a program related problem, and (2) merely reading/copying a solution. This difference with respect to your relationship to the knowledge typically means the difference between failing the exams and passing them.

Learning requires feedback. What forms of feedback should you seek with respect to the assignments in this course? With respect to the programming assignments, you should find the following three sorts of feedback to be beneficial:

1. Syntax
    - Description: The compiler for a particular programming language will check the syntax of your programs. If your program compiles, you know that you have adhered to the syntax of the language.
    - Guidance: Some compilers provide informative error messages with respect to syntax. Rust compilers do just that. Some compilers provide hardly any information with respect to syntax errors, beyond the fact that there is something wrong. Prolog is an example of this sort of compiler. When you need additional help in sorting out compiler errors, you might consult the language specification or various sites online that relate to the language.
2. Correctness
    - Description: If your does not run runs according to specification on an appropriate set of test data, you know that there is something quite wrong with it. This is feedback! If it seems to work on a reasonable range of test data, you should feel moderately confident that it is correct.
    - Guidance: If your program does not run according to specification in some instance, you might want to arrange for it to produce intermediate output either by means of explicit print statements or by means of some trace facility associated with the language. Judicious use of intermediate output is the primary way that one goes about debugging the behavior of a program. The intermediate output is an important form of feedback! Generally speaking, to thoroughly understand the behavior of a program requires that you appreciate the flow of the program, either implicitly (mentally) or explicitly (computationally). If the mental effort doesn't achieve the desired understanding, move on to carefully generating explicit output in a way that makes the flow of execution abundantly clear.
3. Expression
    - Description: A program should be sound with respect to principles an practices of programming. After taking at least CS1 and CS2, you should have a reasonable idea of what it means for a program to be sound with respect to principles and practices of programming. In order to do this, you might like to compare it with other solutions to the problem. You can do this by sampling a relatively small number of solutions to the programming problem from the course web site. Study the better ones, with an eye towards appreciating the significant differences between your solution and the others. Ask yourself pertinent questions with respect to the quality of your program and those in the sample. What do you like more about your program than that of person P? What do you like more about person P's program than yours? What do you like more about person Q's program than person P's program? The answers to these questions is a valuable form of feedback. Moreover, by regularly reading and critiquing programs you will hone your eye for what constitutes sound programming practices and the principles on which they are grounded.

- Guidance: After you have completed your programming assignment in the most authentic manner that you can, perform a parallel terraced scan of some number of your classmates programs, with the goal of identifying a handful of sound programs. Add your own program to the mix. Then, start reading the half dozen programs, and thinking about them, with an eye towards identifying the good, and the not so good, in each – particularly your own! In doing so, you will be generating a wealth of feedback!

With respect to the non-programming assignments, you will have to enter "mechanical mode" to check form (syntax) and function (correctness) with respect to your solutions. Beyond that, you should compare your work to that of your classmates, by browsing the course web site, in order to gain insights into what constitutes good work.

On Real Points and Fake Points

Real points are authentically earned points. Fake points are inauthentically earned points. With respect to attendance and exams, all of the points that you earn in this course will be real points. With respect to the building your web site (and fashioning the assignments that you post to your web site) you have the option of working hard to authentically earn your points, or hardly working to inauthentically earn your points. In a limited, superficial sense, both kinds of points are equal in terms of the final grade. But in a much more significant, vital sense, real points earned on building a web site that represents quality programs that you authored in an authentic fashion, tend to predict success on the exams, and hence the course. Fake points garnered on these activities tend to predict poor performance on the exams, and hence in the course.

Just as fake calories and fake fun can leave you with illusory feelings of satisfaction, fake points in the course can leave you will an illusory notion that you are on track to succeed in the course. Beware! The few additional points that you inauthentically accrue on assignments or the web building activity, above what you might earn through your own hard work, will not compensate for the learning that you miss out on by copying and pasting work that you have not made your own. Furthermore, the false notion of understanding that you may derive from your activities will likely spell doom for you on the exams, which are what really count when it comes right down to your final grade. On the other hand, students who authentically get a substantial majority of the programming and website building points tend to do pretty well on the exams, and in the course.

The assignment and the web site components of the course have been carefully designed to afford you powerful learning opportunities. It is up to you to determine whether or not you will avail yourself of these learning opportunities!

On a somewhat related note, you might enjoy taking a look at the following short text:
`https://emeryberger.medium.com/coping-with-copilot-b2b59671e516`

Important Notes

1. This is an in-person, face-to-face class. I intend to teach the course accordingly, and to adhere to the admonition of my dean:

   - CLAS Dean (August 4, 2021): Faculty should continue to plan to teach their courses as posted in the schedule. **Please don't make individual exceptions to allow students to participate remotely in face-to-face courses. We know how disruptive that is to both faculty and students.**

2. Statement precluding the student use of cell phones or laptops or other electronic communication devices in the classroom: **Students will not be permitted to use cell phones or laptops or Kindles or iPads or Surfaces or other electronic communication devices while class is in session.**

(a) If you should need to check your phone for extraordinary reasons, please just quietly remove yourself from the classroom and check your cell for communications in the hallway. In the case of an emergency or other unexpected exigency, tend to your emergency or unexpected exigency. Otherwise, please simply quietly return to class immediately after checking your phone.

(b) If you are someone who likes to keep your class notes on one of your machines, simply take the best notes that you can by hand, on paper, and then "copy" them after class to your machine, enhancing them in whatever ways you see fit. This activity, in itself, is a great learning activity, one which affords you opportunities to clarify, expand, and reinforce your knowledge.

3. In consideration of lingering COVID consequences, I will generally distribute significant documents, including assignments, by posting them to the course web site.
4. In the event that the college should need to go remote, as it did starting in March, 2020, I plan to use a teaching model that I developed for use at that time and that served me and my students well for the 2+ semesters that in-person teaching was so severely limited. If the need should arise, I will send you an email with a description of the model. With luck, that unfortunate situation will not recur this semester.
5. The course web page is located at:
`http://www.cs.oswego.edu/~blue/course_pages/2022/Fall/Csc416/`
6. Generally speaking, I will be processing student email this semester twice each week, on Monday from 2:00pm-3:00pm and on Friday from 2:00pm-3:00pm. Please expect my response to any email that you may send my way to be timed according to these "student email processing" hours.
7. Generally speaking, I will be holding three office hours each week, on Monday from 7:30am-8:30am, on Tuesday from 9:00am to 10:00am, and Friday from 7:30am-8:30am. I plan to conduct my office hours solely via Google Meet this semester. If you would like to meet with me during an office hour, please send me an email. I will process requests for office hour appointments during my email processing hours from 2:00pm to 3:00pm on Monday and Friday. Generally speaking, if you send me an email request for an office hour appointment by 2pm on a Friday, I will book you for the following Monday morning. If you send me an email request for an office hour appointment by 2pm on Monday, I will book you for the following Tuesday or Friday. If you have a preferred time to meet during my office hours, I will do my best to accommodate your preference. Otherwise, I will schedule on a "first-come/first-served" basis, allocating the next office hour "slot" (15 minutes per slot) that is available.
8. Requests to make up exams will rarely be considered unless accompanied by a written medical excuse for your absence.
9. It is intended that you complete your work by yourself. You are, of course, welcome to ask specific technical questions of others and converse over conceptual issues, but you should be doing your own work. Compelling evidence that someone other than you contributed conspicuously to the completion of required work will result in a "maximum negative" grade for that assignment, failure in the course, or worse.
10. College Intellectual Integrity Statement: SUNY Oswego is committed to Intellectual Integrity. Any form of intellectual dishonesty is a serious concern and therefore prohibited. You can find the full policy online at `http://www.oswego.edu/integrity`.
11. College Disability Statement: "If you have a disabling condition, which may interfere with your ability to successfully complete this course, please contact the Office of Accessibility Services."
12. Clery Act/Title IX Reporting Statement: SUNY Oswego is committed to enhancing the safety and security of the campus for all its members. In support of this, faculty may be required to report their knowledge of certain crimes or harassment. Reportable incidents include harassment on the basis of sex or gender prohibited by Title IX and crimes covered by the Clery Act. For more information about Title IX protections, go to https://www.oswego.edu/title-ix/ or contact the Title IX Coordinator, 405 Culkin Hall, 315-312-5604, titleix@oswego.edu. For more information about the Clery Act and campus reporting, go to the University Police annual report: `https://www.oswego.edu/police/annual-report`.