ECS 140A Programming Languages – Fall 2022
Syllabus

Instructor: Aditya V. Thakur (avthakur@ucdavis.edu)
Teaching Assistants:
Patrick Chapman (pchapman@ucdavis.edu),
Stephanie Nawas (snawas@ucdavis.edu),
Zhe Tao (zhetao@ucdavis.edu)

1   Course Overview

1. Syntax and semantics of programming languages

2. Object-oriented programming and Go

3. Functional programming and Lisp

4. Logic programming and Prolog

5. Concurrent programming in Go

6. Miscellaneous topics such as dynamic data-race detection, garbage collection, etc.

2   Text books

These text books are recommended, but not required.

Kenneth C. Louden, Kenneth Lambert, *Programming Languages: Principles and Practices (3rd Edition)*, Cengage Learning, 2011.

Alan A. A. DonovanBrian W. Kernighan, *The Go Programming Language*, Addison-Wesley, 2015.

William Clocksin, Christopher S. Mellish, *Programming in Prolog: Using the ISO Standard (5th Edition)*, 2003.

Robert Wilensky, *Common LISPcraft (First Edition)*, W. W. Norton & Company, 1986.

3   Online Resources

1. https://go.dev/

2. http://go.dev/tour

3. https://www.golang-book.com/books/intro

4. https://go.dev/doc/effective_go.html

5. https://gobyexample.com/

6. *Concurrency in Go* http://proquest.safaribooksonline.com/9781491941294 (Available online via the UC Davis Library.)

7. *Paradigms of Artificial Intelligence Programming: Case Studies in Common Lisp* https://github.com/norvig/paip-lisp

8. Prolog tutorials and documentation http://www.swi-prolog.org/

4   Lecture Schedule

The lecture schedule is tentative and subject to change.
(M: Main textbook; G: Go textbook; L: Lisp textbook; P: Prolog textbook)

Week	Topic	Reading
1	Overview	M1-2
	Syntax and Parsing	M6-7
2	Semantics	M9.1-9.4
	OO Programming and Go	G1-7
3	OO Programming and Go	
	OO Programming and Go	
4	Functional Programming	M3, L1-9
	Functional Programming	
5	Functional Programming	
	Functional Programming	
6	Logic Programming	M4, P1-4,6-8
	Logic Programming	
7	Logic Programming	
8	Logic Programming	
	Concurrency	G8-9, M13
9	Concurrency	
	Concurrency	
10	Data Race Detection	

5   Grading

5.1   Review Quizzes (10%)

1. These are to be worked on individually.

2. You may refer to the course material when answering the questions.

3. Review quizzes are online quizzes conducted using Canvas.

4. It is the responsibility of the student to ensure that they check Canvas for such quizzes.

5. These quizzes will be open for a week; you can attempt each quiz multiple times and your highest score will be used.

6. The quiz with the lowest score will be dropped.

7. Review Quizzes can be submitted up to 24 hours after the deadline, with a 4% grade penalty applied per hour after the deadline.

8. An extension of the assignment can be requested due to medical reasons. The request has to be made at least 48 hours prior to the assignment deadline. A doctor's note should be provided.

9. Interactive grading will be conducted for a subset of students chosen at random.

5.2   Midterm (20%)

1. The in-class midterm covers the first half of the quarter.

2. Students scoring higher on Final than Midterm will have average of two used in course grade calculation. (Except if SJA referral, see expected conduct).

   Thus, if you scored 50 out of 100 in the midterm and 100 out of 100 in the final exam, then your effective midterm score would be 75 out of 100 (instead of 50).

   However, if you scored 100 of 100 in the midterm and 50 out of 100 in the final exam. then your effective midterm score does not change and would still be 100.

3. Students are allowed to bring notes handwritten on a single A4-sized sheet. Students should write their name on the top-right corner of this sheet, and are required to submit it when turning in the exam.

5.3   Final (30%)

1. The final will be cumulative with emphasis on the second half of the quarter.

2. Students are allowed to bring notes handwritten on a single A4-sized sheet. Students should write their name on the top-right corner of this sheet, and are required to submit it when turning in the exam.

5.4   Programming Assignments (40%)

1. Homework can be worked on in groups of *at most three*. Even if working in a group, each student has to submit a copy of the work to get credit. Moreover, the submission need not be the same for all students in the group. Thus, the submission can be exactly the same as the rest of the group, or a student can make changes in their solution and submit it.

2. There will be five programming assignments, approximately one every other week.

3. Programming assignments can be submitted up to 24 hours after the deadline, with a 4% grade penalty applied per hour after the deadline.

4. An extension of the assignment can be requested due to medical reasons. The request has to be made at least 48 hours prior to the assignment deadline. A doctor's note should be provided.

5. Interactive grading will be conducted for a subset of students chosen at random.

Note: Percent breakdown is subject to change depending upon class performance. You will have one week from the return of your graded work to request a regrade.

6   Expected Conduct

1. You are expected to follow the UC Davis Code of Academic Conduct (CAC):

   http://sja.ucdavis.edu/files/cac.pdf.

2. You are expected to follow the UC Davis Principles of Community (POC):

   http://occr.ucdavis.edu/poc/

6.1   Programming Assignments

1. Discussion of homework and programming assignments outside your group should be limited to general discussion of topics (including with past students).

2. You should form teams on your own; the instructor will not be assigning you to a particular team. You can post a message on Piazza to search for a teammate.

3. You can only be in one group for a particular homework or programming assignment. However, you do not need to be in the same group for all homework and assignments.

4. Viewing of past student submissions is strictly prohibited.

5. You are encouraged to use a private GitHub repository for your assignments, which will help coordinating among group members. Making the solution of any assignment public (even after you have completed the course) is considered academic misconduct.

6. Resubmission of prior work from past courses is strictly prohibited.

7. Discussion of interactive grading with other groups (including past students) is strictly prohibited.

8. All coding projects will be compared with MOSS, including past submissions.

9. You are only allowed to use the subset of the programming languages that we discuss in class. Some of the assignment problems become trivial if you step outside this subset.

6.2   In-class Exams

1. You may not sit next to any student that you are partnering with or have partnered with on any assignment in any class, study with or have studied with regularly, or routinely spend time with socially.

2. Students may receive a zero for violation of expected conduct, even if no academic misconduct found.

6.3   Regrade Requests

1. A regrade request should only point out if a grader stated something specific about your submitted solution that is untrue.

2. The Code of Academic Conduct (CAC) specifically forbids *"Pressuring an instructor or teaching assistant to regrade work, change a final grade, or obtain an exception such as changing the date of an exam, extending a deadline, or granting an incomplete grade."* Such pressure includes (but is not limited to):

   (a) An explicit request to change the final grade.
   (b) Stating that you "you need at least a C in the course" for some reason (say because you are graduating soon).
   (c) Asking what your course grade will be. If the coursework is not all graded, then we do not know the answer to this question. If it is graded, then you can determine the grade yourself.
   (d) Stating that your incorrect solution deserves more partial credit.
   (e) Asking for the grading scale for the course or if a curve will be applied. The default answer is no.
   (f) Pointing out that your score is close to a letter grade cutoff.
   (g) Reacting to a warning about violations of this policy by arguing that it wasn't a violation or that the policy is wrong.

6.4   Online Office Hours

1. Book your appointment with the instructor or TA using the link provided on Canvas.

2. You should book only ONE appointment slot per day.

   This applies for group assignments as well: each group should book only ONE appointment slot per day.

3. Booking multiple office-hour slots in advance is not allowed.

4. Consider posting your question on Piazza; this might get it answered more quickly. Conversely, make sure your question has not already been answered on Piazza before booking an appointment slot.

5. Because there are limited appointment slots, cancel your appointment if you are not able to make it or if your question has already been answered.

6.5   Piazza & Other Communication

1. You are training to be professionals; you are expected to communicate in a professional manner. As a general guideline: if you would not say it to the individuals face, you should not post it.

2. Piazza is not meant as a replacement for attending lectures, discussions, and office hours. The instructors will do their best to answer questions on Piazza. If your question has not been answered it might be because the answer was discussed in class, is stated in the handout, already answered on Piazza, or it is something you are expected to figure out on your own.

3. Do not post code in public forum (only pseudo code), use private posts for help with debugging.

4. Before asking a question, please check that the question has not already been asked.

5. A Piazza post that is not relevant to understanding the course material will be deleted.

6.6   Student Judicial Affairs

1. Matters where collaboration appears to be excessive will be turned over to Judicial Affairs for resolution.

2. Once matters have been submitted to SJA none of the instructional staff may discuss the matter until it is resolved. Do not contact the instructor unless to confess the violation.

3. Those students referred to SJA that are in violation and confess prior to first SJA meeting may possibly receive partial credit on assignment/exam.

4. Those students found in violation that do not confess prior to first SJA meeting will receive zero on the assignment/exam, and cannot receive the midterm/final averaging in course grade calculation.

5. Remember that matters submitted to SJA are for suspicion of violations of student conduct code, innocent students are found to be not in violation all the time.

6. If you are in violation of conduct code and have been referred to SJA, acknowledge your mistake, accept the punishment, learn from it, and let's move on.