CSE396 Course Information Spring 2026

Instructor:

Dr. Kenneth W. Regan, 326 Davis Hall, 645-4738, regan@buffalo.edu

TAs and UTA: (The common Davis 301-302 TA space is used for hours)

Sean Grzenda seangrze@buffalo.edu
Jeffrey Liu jliu294@buffalo.edu

Office Hours:

- Regan, Wed. 1–3pm, Fridays 1–3pm
- Grzenda, Fridays 11am–1pm.

Lectures and Recitations:

(LEC) TuTh 5:00–6:20pm, in Hochstetter 114

(R2) Thursdays 9:30–10:20am, in Clemens 204

(R3) Fridays 10:00–10:50am, in Capen 108

(R4) Fridays 9:00–9:50am, in Capen 108

Examinations:

- Two *prelims*, the first on **Thu., 3/5**, in class period.
- One *cumulative* 3-hr. final.

1 Required Reading

(1) Text: Michael Sipser, *Introduction to the Theory of Computation, 3rd ed.* (Boston: Cengage Learning, 2013). *Intended coverage:* Chapters 0–5 and much of 7 (minus 2.4).

(2) Handouts and course notes provided by the instructors. Some of these will be given out in class besides being posted online.

(3) The course *TopHat* page. There will be online components to some assignments that will be administered from there.

(4) The *CSE396 Piazza page*. Although all official course information will be given in lectures, you may catch it first here, as well as information about the problem sets. Students are invited to post queries of general interest. Please do not, however, post answers unless and until cleared with the instructors and TAs.

(5) There will also be an *Autolab* page, primarily for grading...

(*) The *course webpage* will be for "static" information: syllabus info and notes and posted problem sets and answer keys, material that can be relied on not to change.

2 Organization and Course Policies

Homework will consist solely of weekly problem sets—all due on *Thursdays* (midnight "stretchy") unless circumstances intervene. There are no programming projects or labs. Use of the *Turing Kit* software for certain assignments with "design a machine" problems is optional—as its printjob export can be wonky, the recommended way is to take screenshots (Windows Snipping Tool or similar) and import them.

Assignments must be submitted *as PDF files* (not MS Word files) in order to work with *CSE Autograder*. All submissions must have your *name and recitation* on them. The recitation should be the one that you regularly attend. They must be clearly legible on the PDF.

My (KWR) general policy is that *late work is not acceptable*. In return, you get an answer key shortly afterward, and a relatively quick turnaround of graded work before the next problem set is due. In an exceptional situation, you may contact me *beforehand* about a possible extension.

The course will be graded on a total-points system. Letter grades will also be given for individual exams and some assignments, as a help in telling you where you stand, but only the point totals will have official significance. The weighting of grades in this course is:

Attendance:	3.0%
Homework:	35% (about 10% online, 25% written)
Prelims:	27%
Final:	35%

Instructors reserve the right to 5% leeway in weighting while assigning the final letter grade—this is most typically done for students who do markedly well on the final exam, when it may be treated as if it were worth 40% for that student. This will only be done to an individual student's advantage, and will have no effect on others' grades. Academic honesty and recitation attendance are two prerequisites for this policy to be applicable.

Homework and all exams are designed for a curve with $\geq$ 90% = A, 84–89% = A-, 78–83 = B+, 72–77% = B, 66–71% = B-, 60–65% = C+, 54–59% = C, 48–53% = C-, 42–47% = D+, 36–41% = D, and $<$ 36% = F. Grades will be "curved" only if something systematic happens to make clear that expectations on a problem were misplaced or snow happened or the allocated time was too short. Having course points sum to 667 makes the thresholds for the final course grades into nice, round numbers: 600/667 for an A, 560 for an A-, 520 for a B+, 480 for a B, 440 for a B-, 400 for a C+, 360 for a C, 320 for a C-, 280 for a D+, and 240 for a D and a pass.

2.1 Goal per ABET Criteria

At the end of this course, you should be able to apply mathematical foundations, algorithmic principles, and CS theory to model and design systems—with comprehension of the tradeoffs involved in design choices.

The backbone of the course is the hierarchy of language classes: regular, context-free, polynomial-time, NP, decidable, r.e., and (completely) undecidable. Those represent levels of design and problem-solving. The higher you go in this hierarchy, the more you can express—but the costs of logical analysis and maintenance of your system grow as well.

2.2 Academic Honesty

A university is a *community*, and every community has values and rules that go hand-in-hand with membership in the community. At universities one rule is the standard of *academic honesty* as it has been understood and followed for **all** of the past millennium. This rule is not written down in a standard text such as Magna Carta or the Constitution, but is the same for every educational institution even though they all have individual statements of it. At bottom it is *honesty*. Different cultures and institutions may have different rules and expectations, but all have the principle of abiding by the local rules, and of not misrepresenting the nature of one's work. Students are required to read and abide by the University and CSE statements of the rules:

https://catalogs.buffalo.edu/content.php?catoid=17&navoid=863#academic-integrity

engineering.buffalo.edu/computer-science-engineering/information-for-students/undergraduate-program/cse-undergraduate-academic-policies/cse-academic-integrity-policy.html

In this course, all assignments will be individual, and the *rule* takes a particularly simple form: All assignments must be *your own work*. This term does not need a definition or legal parsing. Tens of thousands have graduated from my *alma mater* with no exam proctors or vetting, just needing to sign "This represents my own work in accordance with University regulations" on every exam paper or major submission. The only difficulty is for those actively (and quite justifiably) seeking help on assignments. The Department has a reasonable guideline tailored to programming projects at the above URL. Note that it is absolute that *writeups* of assignments must be completely your own production—in this course that is especially meaningful because *formal presentation* of solutions is an important course subject and goal in itself. I will talk about "reasonable discussion" of problems in class at some convenient and good time.

Well over half the cheating I used to catch on single problem sets were [*claimed to be*] one student copying off another who couldn't attend class and asked the "trusted friend" to submit for him/her. In such cases the University considers *both* students to be culpable, and my measures reflect this policy. There can be some leniency for "first offenses," but anything more serious is liable for an automatic F in the course and/or explusion from the Department (*cum*-University). The use of *Autograder* obviates this mechanism, but the philosophy remains. The much more serious recent technological development is students sending photos of their work to other students via smartphone—then the guilt of the enabler is sharper.

2.3 Policy on Improper Distribution of Material

All materials prepared and/or assigned by me for this course are for the students' educational benefit. Other than for permitted collaborative work, students may not photograph, record, reproduce, transmit, distribute, upload, sell or exchange course materials, without my prior written permission. This especially goes to say that answer keys—which will be given by non-public links—may not be uploaded to "course helper" sites. But the prohibition applies also to lecture notes (even though posted in public), discussion prompts, study aids, posted sample tests, and any other course-specific presentation materials, packets, or handouts. Violation of this policy may not only subject a student to a finding of "academic dishonesty" under the

Academic Integrity policy and/or disciplinary charges under the Student Code of Conduct, but also constitute copyright infringement in violation of federal or state law..

2.4 Medical Emergencies

Accommodations for medical emergencies will be made on a case-by-case basis. Requests for extensions based on medical emergencies must be accompanied by documentation of the emergency from student health services, for which see

http://www.buffalo.edu/studentlife/who-we-are/departments/health.html.

2.5 Accessibility Resources

If you have any disability which requires reasonable accommodations to enable you to participate in this course, please contact the Office of Accessibility Resources in 60 Capen Hall, 716-645-2608 and also the instructor of this course during the first week of class. The office will provide you with information and review appropriate arrangements for reasonable accommodations, which can be found on the web at: http://www.buffalo.edu/studentlife/who-we-are/departments/accessibility.html.

2.6 Incompletes and Withdrawals

This course shall follow strictly the University's standard for incompletes: they are only to make up work missed owing to circumstances beyond a student's control, and they are given only when a student has enough points to pass the course with already-completed work (i.e., no "I/F" grades). An "Administrative Withdrawal" after the "R" date is available only to students who withdraw from *all* their courses in a given term.

3 Chapters Covered

Part of Chapter 0 is assumed background; some parts will be addressed in the first lecture.

Chapter 1 all—hardcopy notes on the "Myhill-Nerode Theorem" will replace the text's "Pumping Lemma" and do the same examples.

Chapter 2, all except the long half of the proof of the equivalence of CFGs and PDAs, and excepting most of the section 2.4 on "DPDAs" which was introduced in the 3rd edition. (DPDAs will be introduced while covering chapter 3 as a special case of Turing machines.)

Chapter 3 all, with some shortcutting.

Chapter 4 all.

Chapter 5, all except section 5.2 on "PCP," and with more emphasis on mapping reductions.

Chapter 7, all or most, with hardcopy of a shorter version of the *Cook-Levin Theorem*. (Note that Chapter 6 is skipped.)

Sometimes the last lecture includes some topics from later chapters as "FYI" material.