CSC 135 – Computability and Formal Languages

Meetings: The class meeting and final exam times will be found on your Student Center (https://my.csus.edu/). They can also be found at https://www.csus.edu/student-life/class-schedules/ and https://www.csus.edu/academic-affairs/internal/final-exam-schedule.html.

Instructor: Ted Krovetz. Office: 5012 Riverside Hall. Email: tdk@csus.edu. Web: http://krovetz.net/csus. Office Hours: See http://krovetz.net/oh for details.

Texts: We will only use freely available materials. Two free textbooks that will be used occaisionally are Mogensen's Basics of Compiler Design and Heffron's Theory of Computation.

Webpage: Canvas.

Catalog Description: Introduction to limits of computation and techniques for specifying and processing formal languages. Regular languages, regular expressions, finite automata, properties and limitations. Context-free languages, grammars, pushdown automata, properties and limitations. Applications in lexical and syntax analyses, including recursive-descent or table-driven parsing. Turing machines, halting problem, reductions. Introduction to functional programming and related programming language features.

Prerequisite: CSC 28, CSC 35, and CSC 130.

Announcements, Questions and Feedback: Course announcements will be made via Canvas. Questions you have should be made via Piazza (a tool linked from Canvas). Before asking a question, check on Piazza to see if it's already been asked. You should not post anything on Piazza (or other online resources) that gives away too much of a solution -- because this deprives others of finding the solution themselves -- but otherwise students are encouraged to both ask and answer questions on Piazza (or other online resources). Course feedback is also encouraged. If something about the course frustrates you, please let me know. Piazza allows anonymous posting. Questions of a personal nature should be emailed to me directly at tdk@csus.edu.

Homework: Homework will be assigned via the course webpage. A new assignment will appear shortly after each previous one is due. There will be two types of homework, graded and ungraded, each designed to help prepare you for course assessments. Ungraded work can be completed in any way you want. Read carefully the graded homework to know what level of collaboration is allowed. This homework arrangement does not give any personalized feedback on the quality of your work. If you would like more feedback, please come to office hours and ask for it. I am happy to oblige.

Exams: In addition to a final exam there will be three other tests. Two of the tests will require that you bring a wi-fi-connected laptop computer to class that can run on battery power during the exam. If you do not have a suitable laptop, one can be checked-out from the university. Dates are shown on the course schedule, and no make-up exams will be offered except for documented illness. You are allowed a single double-sided page of handwritten notes for each test and final exam.

Some quizzes *may* be given over the semester. If we have any quizzes they will be closed notes, similar to completed homework and will count toward your homework score. Any quiz will be announced at least 72 hours ahead of time via Canvas.

Grading: Graded work is distributed as follows: homework 20%, programming tests 15% each, written test 20%, and final 30%. Your weighted average will determine your grade: A for 90+, A- for 85+, B+ for 80+, B for 75+, B- for 70+, C+ for 65+, C for 60+, C- for 55+, and F for otherwise. These cutoffs are generous, so grades are normally not curved or rounded up to the next grade (even if very close, so please don't ask).

Advice: If you wish to do well in this class try to do all of the following.

(i) Attend class. Although attendance is not taken during lecture, those who skip class usually do less well.
(ii) Do your homework early so that you can get help if needed. Only through practice will you be able to do similar problems on an exam.
(iii) Ask and answer questions in class. One of the best ways to learn is through a dialogue guided by questions and answers. And, you should ask questions when something does not make sense. Do not be afraid -- you will benefit through your participation and the class will be more dynamic and interesting.
(iv) Ask questions outside of class. Office hours and the online forum are usually underused.
(v) Take notes during lecture and while reading. Later copy your notes into a neater and more concise format, making sure you understand everything you are writing.
(vi) Study worked-out solutions when they are supplied and ask questions until you understand.
(vii) Each unit of course credit is supposed to correspond to about three hours per week of time spent on the course. Try to devote at least that much on average.

Academic Integrity: You may complete ungraded homework in whatever way you believe you learn best, alone or in groups. There are no academic integrity issues with regard to ungraded homework.

Graded homework will specify whether collaboration is allowed. If collaboration is not allowed, then you are not allowed to get substantial help from other students in the class. You are only allowed to discuss at a high-level solution strategies. If collaboration is allowed, then you may work with up to two other students, but only if you list each other as collaborators in your submitted work. Any collaboration must be genuine in the sense that an equal amount of work is performed by each member of the collaborative and everyone in it understands fully the result.

If any significant help is found on the internet, you must document your work with the source URL of the online resource. This is good practice for both professional and academic environments.

Regrading: If you believe you lost points on some work even though your solution was correct, contact the instructor within a week of when the work was returned to the class. Explain carefully why you believe your answer is correct. Requests for a regrade made after a week will be denied.

Covid: You should follow CDC guidelines about when to isolate or quarantine. If you need to isolate or quarantine during an assessment, you will be expected to take the assessment remotely at the same time as your classmates, proctored using Zoom with your webcam on. If you are too ill to take an assessment, you must submit documentation from a healthcare provider to be excused from the assessment.

Official Course Outcomes: Students completing this course will be able to

1. Design generators and recognizers for simple regular and context-free languages and give examples of algorithmically converting from one to the other.
2. Demonstrate a context-free grammar ambiguous using parse trees

3. Explain the limits of regular and context-free languages and apply a pumping lemma to show a language not to be regular and/or context-free
4. Discuss parsing and scanning techniques including the principle of longest substring, recursive-descent or table-driven parsing, derivations, parse trees, first and follow sets
5. List the main phases of a compiler, and the function of each
6. Explain functional programming language features and their use, including immutability, higher-order functions, lambda functions, dynamic type systems, and tail recursion
7. Develop solutions to simple problems using a functional programming language
8. Describe the importance of Turing machines, the halting problem, and reductions that prove certain problems cannot be solved by computer algorithms.