CSCI 301: Formal Languages and Functional Programming
Syllabus, Winter, 2017

Instructor: Geoffrey Matthews
Email: `geoffrey dot matthews at wwu dot edu`
Office hours: MTWF 10:00-10:50, CF 469

Lectures:

CRN	Time	Room
11650, 11652	MTWF 9:00-9:50	CF 225
12762, 12763	MTWF 11:00-11:50	AH 014

Lab sessions:

CRN	Time	Room	TA
11650	T 10:00-11:50	CF 414	Katie Hursch
11652	W 10:00-11:50	CF 414	Ellyn Ayton
12762	T 12:00-01:50	CF 416	Elizabeth Brooks
12763	W 12:00-01:50	CF 416	Elizabeth Brooks

You may attend the session you did not sign up for on a space available basis. There are no labs the first and last weeks of class, and the week of the midterm.

Catalog copy: Introduction to discrete structures important to computer science, including sets, trees, functions, and relations. Proof techniques. Introduction to the formal language classes and their machines, including regular languages and finite automata, context free languages and pushdown automata. Turing machines and computability will be introduced. Programming using a functional language is required in the implementation of concepts. Includes lab.

Goals: On completion of this course, students will demonstrate:

1. Thorough understanding of the mathematical definitions of concepts important to computer science, including sets, tuples, lists, strings, languages, graphs, trees, functions and relations
2. The ability to prove basic theorems involving these mathematical concepts
3. The ability to employ effectively the functional programming style in a functional programming language
4. Solid understanding of fundamental classes of languages, including regular and context free, and their corresponding machines
5. Basic understanding of Turing machines and computability
6. Basic understanding of important algorithms, including conversion of finite automata to different forms, conversion of grammars to machines
7. Basic understanding of LL(k) and LR(K) grammars and the parsing techniques used for those grammars

Websites:

- For class materials: `https://github.com/geofmatthews/csci301`
- For turning in homework and grading: `https://wwu.instructure.com/`

Goals: This class is an introduction to computer science *theory*. This is exactly parallel to the distinction between theoretical physics and applied physics. We will use simplified, abstract, ideal mathematical models of computers, so that we can study the theoretical limits of what they can accomplish. We will begin studying the basic mathematics we need, and then study mathematical models of computers of increasing complexity and power. Two classes of computers, finite state automata and push-down automata, also form the basis of powerful programming paradigms useful in a variety of situations.

We will also study the functional programming language *Scheme*. Its simplicity, power, and mathematical elegance will inform our study of computers in the abstract, and also teach us new styles of programming.

Texts:

Math:

- `http://www.people.vcu.edu/~rhammack/BookOfProof/`
- `http://cg.scs.carleton.ca/~michiel/TheoryOfComputation/`
- `http://users.utu.fi/jkari/automata/` Lecture notes.
- Other readings and handouts as they come up.

Scheme:

- `http://www.scheme.com/tspl3/` Note: Use the 3rd edition and do *not* use the 4th edition of this book. Our software is based on R5RS scheme and the 3rd edition covers this. The 4th edition is based on R6RS scheme.
- The *help desk* within Racket leads to documentation on the implementation.
- Optional: `http://mitpress.mit.edu/sicp/`
- Optional: `http://ds26gte.github.io/tyscheme/`
- Other readings and handouts as they come up.

Software: *DrRacket*, available here: `http://racket-lang.org/`

Exams: A midterm and a final, according to the schedule below. The final will be cumulative.

Exams are closed book, with the exception that you may consult two pieces of paper during the exam. You may write or print whatever you wish on these pieces of paper.

Laboratory exercises: Laboratory exercises will be handed out every week. Each lab is due by midnight Tuesday of the following week.

Math work: Math assignments will be handed out as they come up during the quarter. These assignments must be formatted with the LaTeX technical typesetting system and submitted online.

Pedagogically, it is important that you do as much independent work as possible. The assigned homework represents a minimum, but there are many exercises in the books and online to get more practice at the math involved.

I will attempt to grade the homework in a timely fashion. This may necessitate grading only a sample of the problems submitted.

Points will be awarded to each assignment based on its difficulty. Note that there may be an assignment due during dead week.

Assessment and grades: Your grade will be based on the math homework, labs, and the two exams. Weights are as follows.

Please note that Canvas sometimes adds up raw scores and gives you a percentage. This usually does not reflect your weighted score for the class.

Grades will be assigned based on scores as shown. At the discretion of the instructor, scores may be scaled. Awarding ± is at the discretion of the instructor.

Homework	Labs	Midterm	Final
25%	25%	20%	30%

%	90-100	80-89	70-79	60-69	0-60
Grade	A	B	C	D	F

Late work: Submissions are due before midnight of the due date. Anything turned in up to midnight of the following day will be accepted with a 25% penalty. Anything turned in up to midnight of the second day after the due date will be accepted with a 50% penalty. Anything more than 48 hours late will not be accepted. It is your responsibility to make sure all assignments are correctly submitted. If you have trouble submitting on Canvas, please contact your TA or instructor immediately.

Lectures and Reading: The amount of material covered in this class is impossible to absorb with only four hours of lecture a week. It is essential that you go over the assigned readings at least once before class discussion of that material, and attempt to work some of the solved problems.

Many times, class lectures will consist of interactively solving problems, with class participation. Some of the concepts and ideas necessary for solving these problems may not have been discussed previously in lectures. In order to participate actively in the classroom discussion, you will have to read ahead. Assigned readings for each lecture are in the schedule, below. Note that this schedule may change as the quarter progresses.

Attendance policy: Attendance is not required but strongly recommended. Studies show that regular attendance is highly correlated with performance.

You are responsible for all material covered in the lectures, books, handouts, or other assigned reading. If you miss a lecture, make sure you get notes from other students and talk it over with them.

If you have an emergency for one of the exam days, notify me as soon as possible. I will handle each case individually and may, for example, schedule a make-up exam, or adjust your remaining scores to determine your grade.

Academic dishonesty: Please read Appendix D of WWU's Catalog on Academic Dishonesty. It is available online at `http://catalog.wwu.edu`.

Unless specified otherwise, all work for this course is meant to be done **individually.** The work that you turn in for a grade must be completely your own, or you will be guilty of academic dishonesty and could receive an F for the course.

However, it can be a valiable learning experience to discuss work with your fellow students, and this is encouraged. However, after working with a colleague, **you may not keep any paper or electronic copies of anything you produced together!** You may only keep your memories. In particular, this means that **you may not ask for or give help while sitting in front of a computer where the assignment is open!** Also, **you may not use anything a colleague has emailed to you!** Delete the email and do not save a copy.

To help understand what I mean, remember the **Long Term Memory Rule**. You may discuss, sketch, write things down, use your computers, whatever, but after you are done working with your fellow students all files must be deleted, whiteboards erased, and all papers you created must be destroyed. You should then watch a rerun of *the Simpson's*, play a game of ping-pong, take a walk, or something else for half an hour. After this you can go back to your assignment (alone) and use the knowledge you have now gained.

It is very easy for experienced software developers like your instructor and your TA to detect copied assignments. Please do not put us in a situation where we have to fail you for plagiarism.

Schedule:

Wed, Jan 4:	Introduction, explanation of syllabus, expectations of students. How to learn. LaTeX
Fri, Jan 6:	Introduction to Scheme and the Racket environment. *The Scheme Programming Language*, Chapters 1 and 2. **Lab 1 assigned.**
Mon, Jan 9:	Continue introduction to Scheme and Lab 1. Sets. *Book of Proof*, Chapter 1.
Tue, Jan 10:	Logic. *Book of Proof*, Chapter 2.
Wed, Jan 11:	Counting, *Book of Proof*, Chapter 3.
Fri, Jan 13:	More Scheme. *Structure and Interpretation of Computer Programs*, Chapter 1. **Lab 2 assigned.**

Mon, Jan 16:	MLK day, no classes.
Tue, Jan 17:	**Lab 1 due.**
	Direct proof, *Book of Proof*Chapter 4.
	Contrapositive Proof, *Book of Proof*, Chapter 5.
Wed, Jan 18:	Proof by contradiction, *Book of Proof*, Chapter 6.
	Proving non-conditional statements. *Book of Proof*, Chapter 7.
Fri, Jan 20:	More Scheme. *Structure and Interpretation of Computer Programs*, Chapter 2.
	Proofs involving sets, *Book of Proof*, Chapter 8.
	Disproof. *Book of Proof*, Chapter 9.
	Lab 3 assigned.
Mon, Jan 23:	Mathematical induction. *Book of Proof*, Chapter 10.
Tue, Jan 24:	**Lab 2 due.**
	Relations and Functions. *Book of Proof*, Chapters 11, 12.
Wed, Jan 25:	Cardinality of Sets. *Book of Proof*, Chapter 13.
Fri, Jan 27:	**Lab 4 assigned.**
Mon, Jan 30:	Deterministic finite automata.
	The Theory of Computation, 2.1, 2.2.
	Automata and Formal Languages Lecture Notes, 2.1.
Tue, Jan 31:	**Lab 3 due.**
	Nondeterministic finite automata.
	The Theory of Computation, 2.4, 2.5.
	Automata and Formal Languages Lecture Notes, 2.2, 2.3.
Wed, Feb 1:	Regular expressions, closure properties.
	The Theory of Computation, 2.7, 2.8.
	Automata and Formal Languages Lecture Notes, 2.4, 2.6.
	`http://cs.stackexchange.com/questions/2016/how-to-convert-finite-automata-to-regular-`
Fri, Feb 3:	The pumping lemma for regular languages.
	The Theory of Computation, 2.9.
	Automata and Formal Languages Lecture Notes, 2.5.
No lab this week.	
Mon, Feb 6:	Myhill-Nerode theorem.
	Automata and Formal Languages Lecture Notes, 2.8.
Tue, Feb 7:	**Lab 4 due.**
	Review.
Wed, Feb 8:	**Midterm exam.**
Fri, Feb 10:	Review midterm
	Lab 5 assigned.
Mon, Feb 13:	Context-free grammars.
	The Theory of Computation, 3.1, 3.2, 3.3.
	Automata and Formal Languages Lecture Notes, 3.1, 3.2.
Tue, Feb 14:	Chomsky normal form.
	The Theory of Computation, 3.4.
	Automata and Formal Languages Lecture Notes, 3.3.
Wed, Feb 15:	Pushdown automata.
	The Theory of Computation, 3.5, 3.6.
	Automata and Formal Languages Lecture Notes, 3.4.
	Equivalence of PDA and CFG. LL parsing.
	The Theory of Computation, 3.7.
Fri, Feb 17:	**Lab 6 assigned.**
Mon, Feb 20:	President's day, no classes.
Tue, Feb 21:	Pumping lemma for CFL.
	The Theory of Computation, 3.8.
	Automata and Formal Languages Lecture Notes, 3.5.

	Closure properties for CFL.
	Automata and Formal Languages Lecture Notes, 3.6.
Fri, Feb 24:	**Lab 7 assigned.**
Mon, Feb 27:	Decision algorithms for regular languages.
	Automata and Formal Languages Lecture Notes, 2.7.
	Decision algorithms for context free languages.
	Automata and Formal Languages Lecture Notes, 3.7.
Tue, Feb 28:	**Lab 6 due.**
	Turing machines.
	The Theory of Computation, 4.1, 4.2, 4.3, 4.4.
	Automata and Formal Languages Lecture Notes, 4.1, 4.2, 4.3, 4.4.
Wed, Mar 1:	Decidability
	The Theory of Computation, 5.1, 5.2, 5.3.
	Automata and Formal Languages Lecture Notes, 4.5, 4.6, 4.7.
Fri, Mar 3:	Enumerability
	The Theory of Computation, 5.4, 5.5, 5.6, 5.7, 5.8.
No lab this week.	
Mon, Mar 6:	Review.
Tue, Mar 7:	**Lab 7 due.**
	Review.
Wed, Mar 8:	Review.
Fri, Mar 10:	Review.
Tue, Mar 14:	Final Exam, 11:00am lecture, 8:00-10:00am
Thu, Mar 16:	Final Exam, 9:00am lecture, 8:00-10:00am