

📧 Everything on this syllabus is subject to change as the semester progresses.

*Class Meetings:* MW 5:00-6:15pm online

*Instructor:* Jay Ligatti ([ligatti@usf.edu](mailto:ligatti@usf.edu)) (For office hours: email for an online appt.)

*Teaching Assistant:* Kevin Dennis ([kevindennis@mail.usf.edu](mailto:kevindennis@mail.usf.edu)) (Also email for an online appt.)

*TA Duties Include:* Grading assignments and answering questions about assignments

*Recommended Textbook:* Compiler Construction: Principles and Practice by K. Louden

*Prerequisite:* EEL 4851C or COP 4530

*URLs:* Please check the course webpage (<http://www.cse.usf.edu/~ligatti/compilers/21>) regularly for announcements, assignments, and an up-to-date schedule. You will also use Canvas (<http://my.usf.edu/>) to upload programming assignments, see course grades, and attend class.

*Attendance:* To attend class, click the Blackboard Collaborate Ultra link in the Canvas page for this course and join the session. I don't take attendance in class, but absences put you at risk for missing assignments, schedule updates, and material not covered in the textbook.

Most of the class meetings will be driven by student questions or comments. When you want to enter the discussion, please use your judgment regarding whether to raise your hand in Blackboard Collaborate or simply unmute yourself and begin speaking. If too many people seem to be speaking at once, I'll try to moderate the discussion and ask participants to raise their hands before joining the discussion. I apologize for mispronouncing your names when calling on you to speak.

Please do not record class lectures in any way, including taking screenshots or audio or video recordings. This policy is intended to respect everyone's privacy and create an open atmosphere for dialog.

I will email any notes I type during class meetings, so you don't have to spend class time typing what I'm typing.

*Course Description:* Introduction to techniques for compiling software; lexical, syntactic, and semantic analyses; abstract syntax trees; symbol tables; code generation and optimization.

**This is a project-based course. The goal is for you, *working by yourself*, to implement your own compiler.** This course has no tests or quizzes.

*Assignments:* There will be six programming assignments, due by 11:59pm (23:59:00) on the following dates: 01/24, 02/07, 02/28, 03/14, 04/04, and 04/25. These assignments will build a compiler for programs written in a new language called DJ (Diminished Java). Do not post your assignment solutions on any medium that could be accessed by other current or future Compilers students (e.g., in a public GitHub repository), as doing so may make you an accessory to another student's plagiarism.

*Late Assignments:* For each day an assignment is late—up to a maximum of 5 days—the grade is reduced 10%. For example, if you submit a 90%-correct assignment 2 days late, your overall assignment score will be 70%.

*(Non-)Group Work Policy:* Everything you turn in for this course must be your own, individual work. Except in very high-level, implementation-independent terms, do not discuss the assignments outside of class and office hours. Class time will be used to discuss the assignments and address any questions that arise.

*Final-Grade Breakdown:*

15%	Assignment I
5%	Assignment II
15%	Assignment III
10%	Assignment IV
25%	Assignment V
30%	Assignment VI
-----	
100%	Total

*Grading Scale:* The scale for final letter grades is as follows, using standard notation for ranges: A ( $\infty, 93.3]$  A- (93.3,90] B+ (90,86.7] B (86.7,83.3] B- (83.3,80] C+ (80,76.7] C (76.7,73.3] C- (73.3,70] D+ (70,66.7] D (66.7,63.3] D- (63.3,60] F (60,0]. An A+ may be awarded for exceptionally outstanding work.

*Email:* For questions related to the course material, schedule, or grading, please first email the teaching assistant. If you have done so but are not satisfied with the response, please email the instructor. Allow at least 48 hours for a response.

*Academic Honesty:* Again, everything you turn in for this course must be your own, individual work. Students caught violating academic integrity will receive an FF grade for the course.

Many additional USF policies (e.g., regarding academic integrity and COVID-19) may be accessed at: <https://www.usf.edu/provost/faculty/core-syllabus-policy-statements.aspx>

***Tentative Schedule***

<u>Week</u>	<u>Dates</u>	<u>Topics</u>	<u>Recommended Reading</u>
1	01/11, 01/13	Introduction; Compilation phases; DJ	1.1-1.7
2	01/20	Lexical analysis	2.1, 2.3
3	01/25, 01/27	Lexical analysis	2.2, 2.4, 2.6
4	02/01, 02/03	Syntactic analysis	3.1-3.4, 5.1-5.2
5	02/08, 02/10	Syntactic analysis	5.3-5.5, 4.3.1-4.3.2
6	02/15, 02/17	Syntactic analysis; Abstract syntax trees	4.1-4.2, 4.3.3
7	02/22, 02/24	Abstract syntax trees	
8	03/01, 03/03	Semantic analysis	
9	03/08, 03/10	Semantic analysis	
10	03/15, 03/17	Code generation	7.1, 7.3.1, 7.4
11	03/22, 03/24	Code generation	
12	03/29, 03/31	Code generation	
13	04/05, 04/07	Code generation	
14	04/19, 04/21	Code optimizations; Bootstrapping	8.9
15	04/26, 04/28	Q&A	