ITP 439 – Compiler Development
Units: 4
Spring 2020—M/W—Time: 5-6:50PM

Location: KAP 163

Instructor: Sanjay Madhav
Office: OHE 530H
Office Hours: T/Th 2-4:15PM
Contact Info: All general course/assignments questions should be asked on Piazza (every student will receive an invitation at the start of the semester).
Personal questions should be directed via email at madhav@usc.edu.

Teaching Assistants: TBD
Office: TBD
Office Hours: TBD
Contact Info: Via Piazza.

Course Description
Students will learn about practical applications of techniques used to develop a programming language compiler. Topics covered include how to parse the program's text, represent the program in memory, optimize the code, and generate machine code. Throughout the semester, students will implement components of a working compiler for a subset of C called University Simple C, which was created specifically for this course. This compiler is implemented in C++. Students will also use the extremely popular and industry-standard LLVM framework for many aspects of the compiler.

Learning Objectives
- Gain a fundamental understanding of how compilers work, with a focus on optimization and the back end
- Implement several aspects of a working compiler for a C programming language subset
- Learn how to work with an existing large-scale code base (the LLVM library)

Prerequisite(s): CSCI 104 or ITP 365. Junior standing or higher recommended.

Course Notes
Lecture slides and assignments will all be posted on Blackboard. Course discussions will occur on Piazza. Assignments will be submitted through BitBucket.

Technological Proficiency and Hardware/Software Required
Students should have access to their own computer running either Windows, MacOS, or Linux, and should be familiar with the basic operation of their computer.

Required Readings and Supplementary Materials
Cooper, Keith and Linda Torczon. *Engineering a Compiler (Second Edition)*. Morgan Kaufman. 2012. ISBN-10: 012088478X.

Additional readings such as excerpts from other books or research papers will be provided on Blackboard.

Description and Assessment of Assignments
There are six different C++ programming assignments in this course, and students will have two or three weeks per programming assignment. The assignments are as follows:

1. Recursive descent parser
2. Semantics and Symbol Table Generation
3. Generation of LLVM IR
4. Static Single Assignment form
5. Optimization passes
6. Register allocation via graph coloring

Programming assignments each have a suite of unit tests. Student grades are based on the percentage of unit tests the assignment passes.

Additionally, there are five short problem sets assigned in between these programming assignments. These problem sets both reinforce concepts and help prepare for the exam.

Exams
There is a midterm and final exam. The final exam is cumulative.

Grading Breakdown

Item	% of Grade
Programming Assignments (6% each)	36
Problem Sets (3% each)	15
Midterm Exam	24
Final Exam	25
Total	100

Grading Scale (Example)
Course final grades will be determined using the following scale
A   93-100
A-   90-92
B+   87-89
B   83-86
B-   80-82
C+   77-79
C   73-76
C-   70-72
D+   69
D   67-69
D-   66
F   65 and below

Half percentage points will be rounded up to the next whole percentage. For instance, 89.5% is an A-, but 89.4% is a B+.

Assignment Submission Policy
Programming assignments must be submitted to student's BitBucket repositories by 11:59PM of the deadline date or will be considered late. Programming assignments that do not compile will receive a 0. Problem sets will be due at the start of class on the designated due date.

Grading Timeline
Students will receive grades on programming assignments and problem sets within one week after the due date.

Late Policy for Programming Assignments
Students are given four "slip" days for the entire semester. These slip days can be used one at a time or all at once. Upon consumption of these slip days, students will be assessed a 25% penalty per day late, for up to three additional days. If a student uses all his or her slip days, extensions will only be given in the event of a documented emergency.

Late Policy for Problem Sets
Problem sets are due at the start of class on the designated due date. They will not be accepted late, barring a documented emergency.

Make-up Policy for Exams
To make up for a missed exam, the student must provide a satisfactory reason (as determined by the instructor) along with documentation. Make-up exams are only allowed under extraordinary circumstances.

Grading Issues

Students will have two weeks after graded feedback is given to contest scores (e.g. assignments and exams). After two weeks, scores will not be changed.

Plagiarism and Individual Work Policy

In this class, programming assignments are expected to represent the individual effort of each student. All programming assignment submissions will be compared with current, previous, and future students' submissions using MOSS, which is a code plagiarism identification program. If your code significantly matches another student's submission, you will be referred to SJACS with a recommended penalty of an F in the course.

It is okay to discuss solutions to specific problems with other students, but it is not okay to look through another student's code. It does not matter if this code is online or from a student you know, it is cheating. Do not share your code with anyone else in this or a future section of the course, as allowing someone else to copy your code carries the same penalty as copying the code yourself.

Course Material Policy

Do not reproduce, distribute, or post any lecture material, assignments, assignment solutions, or exams publicly without written consent of the instructor. You may take notes and make copies of course materials for your own use. You may not post course materials on sites like CourseHero. Doing so is a copyright violation and in some cases may also be an academic integrity violation that will be dealt with accordingly

Course Schedule

	Lecture Topics	Readings/Due Dates
1/13	Intro; Compiler Basics; Scanning	Ch. 1; §2.1-2.3
1/15	Top-down parsing	§3.1-3.3
1/20	*MLK Day (no class)*	
1/22	Bottom-up parsing; Semantic Analysis Basics	§3.4; §4.1-4.2; §5.5; **PS1 DUE start of class**
1/27	Context Sensitive/Attribute Grammars	§4.3-4.4
1/29	IR; Generating Expressions; Control Flow	§5.1-5.3; §7.1-7.4; §7.8; **PA1 DUE 1/31 @ 11:59PM**
2/3	Arrays, Strings, Calling Conventions	§6.1-6.3; §7.5-7.7; §7.9
2/5	OOLs; Basics of Optimization	§6.4-6.6; §8.1-8.4; **PS2 DUE start of class**
2/10	Basics of Data-flow Analysis; Dominators	§9.1-9.2.1
2/12	Available Expressions; Live-variable analysis	§9.2.2; **PA2 DUE 2/14 @ 11:59PM**
2/17	*President's Day (no class)*	
2/19	SSA (Cytron Algorithm); Paper Discussion	§9.3.4-9.3.6; **PS3 DUE start of class**
2/24	Midterm Review	
2/26	*__Midterm Exam__*	
3/2	Static Single Assignment, Part II	"Simple and Efficient SSA Construction"
3/4	Optimizing Compiler Case Studies	**PA3 DUE 3/6 @ 11:59PM**
3/9	TBD	
3/11	TBD	
	Spring Break	
3/23	Code Motion; LLVM Opt Passes	§10.3; "Writing LLVM Opt Passes"
3/25	Intraprocedural Analysis; Aliasing	**PA4 3/27 DUE @ 11:59PM**
3/30	The Backend; Instruction Selection	§11.1-11.4;
4/1	Instruction Selection, Part II; Paper Discussion	§11.5-11.7; **PS4 DUE start of class**
4/6	Global Register Allocation	§13.1-13.4;
4/8	Register Allocation, Part II	**PA5 DUE 4/10 @ 11:59PM**
4/13	Instruction Scheduling	§12.1-12.3
4/15	Instruction Scheduling, Part II	§12.4-12.6
4/20	Paper Discussion	**PS5 DUE start of class**
4/22	Vectorization Techniques	"Program Optimization Through Loop Vectorization"
4/27	Security and Compilers	"Compilers and Software Security"
4/29	Final Exam Review	**PA6 DUE 4/31 @ 11:59PM**
FINAL	**Final exam according to final exam schedule**: Wednesday, May 6, 4:30-6:30PM	

For due dates, "PS" is a Problem Set and "PA" is a Programming Assignment.

Assigned readings in quotes will be available either directly or via link on Blackboard.

Statement on Academic Conduct and Support Systems

Academic Conduct

Plagiarism – presenting someone else's ideas as your own, either verbatim or recast in your own words – is a serious academic offense with serious consequences. Please familiarize yourself with the discussion of plagiarism in *SCampus* in Part B, Section 11, "Behavior Violating University Standards" https://policy.usc.edu/scampus-part-b/. Other forms of academic dishonesty are equally unacceptable. See additional information in *SCampus* and university policies on scientific misconduct, http://policy.usc.edu/scientific-misconduct.

Support Systems

Student Counseling Services (SCS) - (213) 740-7711 – 24/7 on call
Free and confidential mental health treatment for students, including short-term psychotherapy, group counseling, stress fitness workshops, and crisis intervention. https://engemannshc.usc.edu/counseling/

National Suicide Prevention Lifeline - 1-800-273-8255
Provides free and confidential emotional support to people in suicidal crisis or emotional distress 24 hours a day, 7 days a week. http://www.suicidepreventionlifeline.org

Relationship and Sexual Violence Prevention Services (RSVP) - (213) 740-4900 - 24/7 on call
Free and confidential therapy services, workshops, and training for situations related to gender-based harm.
https://engemannshc.usc.edu/rsvp/

Sexual Assault Resource Center
For more information about how to get help or help a survivor, rights, reporting options, and additional resources, visit the website: http://sarc.usc.edu/

Office of Equity and Diversity (OED)/Title IX Compliance – (213) 740-5086
Works with faculty, staff, visitors, applicants, and students around issues of protected class.
https://equity.usc.edu/

Bias Assessment Response and Support
Incidents of bias, hate crimes and microaggressions need to be reported allowing for appropriate investigation and response. https://studentaffairs.usc.edu/bias-assessment-response-support/

The Office of Disability Services and Programs
Provides certification for students with disabilities and helps arrange relevant accommodations. http://dsp.usc.edu

Student Support and Advocacy – (213) 821-4710
Assists students and families in resolving complex issues adversely affecting their success as a student EX: personal, financial, and academic. https://studentaffairs.usc.edu/ssa/

Diversity at USC
Information on events, programs and training, the Diversity Task Force (including representatives for each school), chronology, participation, and various resources for students. https://diversity.usc.edu/

USC Emergency Information
Provides safety and other updates, including ways in which instruction will be continued if an officially declared emergency makes travel to campus infeasible, http://emergency.usc.edu

USC Department of Public Safety – 213-740-4321 (UPC) and 323-442-1000 (HSC) for 24-hour emergency assistance or to report a crime.
Provides overall safety to USC community. http://dps.usc.edu