



1 Website

<https://www.swamiiyer.net/cs451/> 


2 Course Description

Introduction to compiler organization and implementation, including formal specifications and algorithms for lexical and syntactic analysis, internal representation of the source program, semantic analysis, run-time environment issues and code generation. Students will write a compiler for a reasonably large subset of a contemporary language, targeted to a virtual machine.

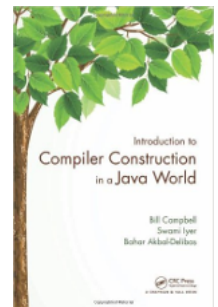
Prerequisites: CS310  and CS420  or CS622 ; or permission of the instructor.

This course meets program outcomes  1a (demonstrate proficiency in problem solving, software design and development, data structures, algorithms, computer organization and computer architecture) and 1b (apply this proficiency to larger mid-sized systems such as compilers). Students who successfully complete this course will be able to: write parsers and produce an abstract syntax tree (AST); analyze and generate code for a programming construct represented by an AST; and allocate physical registers (a limited resource) to a program expressed in terms of virtual registers (an unlimited resource).

3 Text

Introduction to Compiler Construction in a Java World  by Bill Campbell, Swami Iyer, and Bahar Akbal-Delibas

The text enables a deep understanding of the Java programming language and its implementation. It covers all of the standard compiler topics, including lexical analysis, parsing, abstract syntax trees, semantic analysis, code generation, and register allocation. It also demonstrates how JVM code can be translated to a register machine, specifically the MIPS architecture. In addition, the text covers recent strategies, such as just-in-time compiling and hotspot compiling.



4 Topics Covered

- Course Mechanics
- Programming Environment
- Compilation
- Lexical Analysis
- Parsing
- Type Checking
- JVM Code Generation
- Translating JVM Code to MIPS Code
- Register Allocation

5 Grading

Students' final grades are determined as follows:

Assessment	% of Final Grade
Projects (1, 2, 3, 5, and best of 4 and 6)	25
Exams (1 and 2) [†]	70
Participation	5

[†] If you score at least 80% on both exams, the higher of the two scores will be considered as your exam average.

The projects:

#	Project
1	Supporting Simple Operations
2	Scanning
3	Parsing
4	Scanning and Parsing with JavaCC
5	Type Checking and Code Generation
6	Register Allocation

6 Academic Honesty

Cheating on the projects or exams constitutes a violation of the academic honesty code, and will be handled according to the procedures delineated in the Student Code of Conduct, Appendix B [↗](#).

7 Accomodations for Students with Disabilities

Section 504 of the Americans with Disabilities Act of 1990 offers guidelines for curriculum modifications and adaptations for students with documented disabilities. If applicable, students may obtain adaptation recommendations from the Ross Center for Disability Services [↗](#). The student must present these recommendations and discuss them with the instructor within a reasonable period, preferably by the end of Add/Drop period.