



CS 516: Compiler Design & Implementation

SAMPLE SYLLABUS

Instructor: Eric Koskinen
Course Web Address: See Canvas
Prerequisite(s): CS334 and (CS590 or CS570 or CS385 or CS182)

DESCRIPTION

In the modern software industry, there is a proliferation of programming languages, especially domain-specific languages, such as Facebook Hack, Google Go, and Mozilla Rust. Therefore, the robustness and security of the software of the future depends on these languages (i) being built from classical compilers foundations and (ii) increasingly incorporating better analysis techniques. This course introduces students to the structure and design of compilers (lexical and syntax analysis, symbol table construction, code generation, etc.) as well as semantic analyses for correctness and security. The emphasis in this course is on the integration of the various parts of a compiler. Students will build a compiler for a small language.

LEARNING OBJECTIVES

After successful completion of this course, students will be able to:

- Provide all students with experience building a significant software project.
- Prepare undergraduate students for industry, where there are increasingly many compilers for DSLs.
- Prepare undergraduate student for graduate work in the fields of programming languages, systems, and/or security.
- Provide foundational material and a relevant programming project for graduate students working in the fields of programming languages, systems, and/or security.
- Build on Automata (CS334) and Algorithms (CS385 or CS570 or CS590). Synergize with Programming Languages (CS496 or CS510).

MATERIALS

Textbook(s):
1. E Compilers (2nd edition) by Aho, Lam, Sethi, and Ullman
2. Advanced Compiler Design and Implementation by Muchnick
3. Modern Compiler Implementation in ML by Appel

Other Readings: See Canvas.

Materials: To be given in class.

REQUIREMENTS

Attendance	Students are required to attend all classes.
Projects	There will be six (6) homework (programming) project throughout this course. Code that does not compile will not be accepted. See below for policies regarding submissions. Policy for late submissions:
Exams	There will be two exams in this course: a Midterm and a Final. Exams are cumulative, but focus on new material. Exam dates are listed in the <i>Tentative Course Schedule</i> available in Canvas.

COURSE POLICIES AND GRADING PROCEDURES

There are 100 possible points that a student can earn in this course. Percentages are listed below.

Homework	(70%)
Midterm	(12%)
Final Exam	(18%)

General Project Guidelines

- Projects that do not compile will get no credit.
- Projects that compile will receive partial credit according to the grading guidelines provided in the project description.
- **Groups:** Teams for group projects will consist of two students.
- **Difficulty:** The course projects are *challenging and time consuming*. They may require significant effort to understand the implementation details and concepts: **Start Early!**
- **Late Projects:**
 - All projects are due at 11:59pm on the specified date.
 - Projects submitted late, but within 24 hours of the deadline are subject to a 10 point penalty.
 - Projects submitted late, but between 24 and 48 hours of the deadline are subject to a 20 point penalty.
 - If you have an extremely unusual circumstance, such as a serious and extended illness, contact the course instructor for special consideration.
 - If, for some reason, you have an extension of less than 48 hours on the homework, the extension only overrides the associated late penalty – it does not extend the late period!
 - **Regrade:** Regrade requests should be sent to the TA. Only reasonable requests will be considered. The entire project or exam will be regraded. Note that this means that the score on a regraded project might decrease.

TENTATIVE COURSE SCHEDULE

Lec.	Topics	Assignments Due
1	Introductions: Compilers, Interpreters, OCaml	HW1: Hellocaml
2	OCaml Crash Course: Translating Simple to OCaml	
3	X86lite	
4	X86lite programming / C calling conventions / Direct AST to X86lite	
5	Intermediate Representations I	
6	Intermediate Representations II	HW2: X86lite
7	LLVM	
8	Structured Data in the LLVM IR	
9	Lexing: DFAs and ocamllex	HW3: LLVMlite
10	Parsing I: Context Free Grammars	
11	Parsing II: LL(k) parsing	
12	Parsing III: LR Grammars and Shift/Reduce	
13	Parsing IV: LR(0) Shift/Reduce DFA	
14	Parsing V: LR(1) Parsing	
15	Parsing VI: Menhir and Oat	
16	First-class Functions	HW4: Compiling Oat v.1
17	Types I: Scope	
18	Types II: Judgments and Derivations	
19	Closure Conversion	
20	Types III: Subtyping, OAT Types	HW5: Full Oat
21	Compiling Objects and Classes	
22	Multiple Inheritance	
23	Optimizations I	HW6: Dataflow Analysis and Register Allocation
24	Data Flow Analysis	
25	Register Allocation	
26	Data Flow Analysis II	
27	Control Flow Analysis	
28	SSA Revisited	

ACADEMIC INTEGRITY

Undergraduate Honor System

Enrollment into the undergraduate class of Stevens Institute of Technology signifies a student's commitment to the Honor System. Accordingly, the provisions of the Stevens Honor System apply to all undergraduate students in coursework and Honor Board proceedings. It is the responsibility of each student to become acquainted with and to uphold the ideals set forth in the Honor System Constitution. More information about the Honor System including the constitution, bylaws, investigative procedures, and the penalty matrix can be found online at <http://web.stevens.edu/honor/>

The following pledge shall be written in full and signed by every student on all submitted work (including, but not limited to, homework, projects, lab reports, code, quizzes and exams) that is assigned by the course instructor. No work shall be graded unless the pledge is written in full and signed.

"I pledge my honor that I have abided by the Stevens Honor System."

Reporting Honor System Violations. Students who believe a violation of the Honor System has been committed should report it within ten business days of the suspected violation. Students have the option to remain

anonymous and can report violations online at www.stevens.edu/honor.

Individual and Group Projects

For individual projects and group projects, the following guidelines should be followed:

- **For individual projects, you must type in and edit your own code, documentation, and any other materials submitted for grading.**
 - Copying someone else's file is not allowed.
 - Allowing someone else to copy a file of yours, either explicitly or implicitly by leaving your code unprotected, is not allowed.
 - Editing each other's files is not allowed.
- **Regarding the ethics of what you may or may not discuss with others:**
 - **"High level" discussions are fine.** For example, discussions about the problem statement.
 - **"Low level" discussions are fine.** For example, discussions about OCaml syntax or using omake, understanding compiler error messages, understanding the mechanics of the tools and libraries used for the projects.
 - **"Mid level" discussions require discretion.** In this CS course, discussions at this level must be limited. Unless explicitly stated otherwise, you may not collaborate significantly with classmates (except group project members) at this level. If you have minor discussions with others at this level or get help from outside resources (tutors, web sites, etc), you must cite in the readme of the submitted projects the names of the people or websites who helped you and how they did. For example:

```
Name: Chris Brown
Email: ***@stevens.edu
I received tips from Jo Johnson on the i/o and example.com/mem.htm on memory
...
```
- **If there is any doubt about the use of external sources or collaboration, please ask for clarification by the course staff.**
- Although we encourage you to use version control software (e.g. GIT) for the course projects, please **do not** distribute the solutions.

EXAM ROOM CONDITIONS

The following procedures apply to exams for this course. As the instructor, I reserve the right to modify any conditions set forth below by printing revised Exam Room Conditions on the exam.

1. Students may use the following devices during quizzes and/or exams. Any electronic devices that are not mentioned in the list below are not permitted.

Device	Permitted?	
	Yes	No
Laptops		x
Cell Phones		x
Tablets		x
Smart Watches		x
Google Glass		x
Other (specify)		x

2. Students may use the following materials during quizzes and/or exams. Any materials that are not mentioned in the list below are not permitted.

Material	Permitted?	
	Yes	No
Handwritten Notes		x
Typed Notes		x
Textbooks		x
Readings		x

3. Students are *not* allowed to work with or talk to other students during quizzes and/or exams.

LEARNING ACCOMODATIONS

Stevens Institute of Technology is dedicated to providing appropriate accommodations to students with documented disabilities. Student Counseling and Disability Services works with undergraduate and graduate students with learning disabilities, attention deficit-hyperactivity disorders, physical disabilities, sensory impairments, and psychiatric disorders in order to help students achieve their academic and personal potential. They facilitate equal access to the educational programs and opportunities offered at Stevens and coordinate reasonable accommodations for eligible students. These services are designed to encourage independence and self-advocacy with support from SCDS staff. The SCDS staff will facilitate the provision of accommodations on a case-by-case basis. These academic accommodations are provided at no cost to the student.

Disability Services Confidentiality Policy

Student Disability Files are kept separate from academic files and are stored in a secure location within the office of Student Counseling, Psychological & Disability Services. The Family Educational Rights Privacy Act (FERPA, 20 U.S.C. 1232g; 34CFR, Part 99) regulates disclosure of disability documentation and records maintained by Stevens Disability Services. According to this act, prior written consent by the student is required before our Disability Services office may release disability documentation or records to anyone. An exception is made in unusual circumstances, such as the case of health and safety emergencies.

For more information about Disability Services and the process to receive accommodations, visit <https://www.stevens.edu/sit/counseling/disability-services>. If you have any questions please contact: Lauren Poleyeff, Psy.M., LCSW - Disability Services Coordinator and Staff Clinician in Student Counseling and Disability Services at Stevens Institute of Technology at lpoleyef@stevens.edu or by phone (201) 216-8728.

INCLUSIVITY STATEMENT

Stevens Institute of Technology believes that diversity and inclusiveness are essential to excellence in education and innovation. Our community represents a rich variety of backgrounds, experiences, demographics and perspectives and Stevens is committed to fostering a learning environment where every individual is respected and engaged. To facilitate a dynamic and inclusive educational experience, we ask all members of the community to:

- be open to the perspectives of others
- appreciate the uniqueness of their colleagues
- take advantage of the opportunity to learn from each other
- exchange experiences, values and beliefs
- communicate in a respectful manner
- be aware of individuals who are marginalized and involve them
- keep confidential discussions private