

Software Atelier 5

Languages and Compilers

Fall 2012

Administrivia:

Instructor: Nate Nystrom
Office: SI-203
Office Phone: +41 (0)58 666 43 10 (do not use)
Email: nate.nystrom@usi.ch (use rarely)

Assistants: Francesco Alberti, Andrea Mattavelli

Staff email: usi-compilers@googlegroups.com

Class Time: Monday, Wednesday, Friday, 13:30–15:15
Class Location: SI-003

Web: <http://inf.usi.ch/nystrom/teaching/compilers/fa12>
Moodle: <http://www2.icorsi.ch/course/view.php?id=132>

Objective: In this course you will learn how programs written in high-level languages are executed on modern hardware. Understanding how languages are implemented is useful for reasoning about program behavior and performance. A secondary goal of the course is to expose students to the principles, techniques, and tools used to construct compilers and interpreters.

Content: The course will cover both the theory and practice of programming language implementation. Topics include compiler structure, lexical and syntactic analysis (parsing), types, semantic analysis, program representations, data-flow analysis, register allocation, optimization, and compiler construction tools. Students will implement a compiler for a high-level programming language on modern hardware.

Teaching mode: The course is heavily project-based. Students will implement a compiler for a high-level programming language on modern hardware. Students will also do written assignments to understand programming language and compiler principles. Students are expected to read the assigned readings before class. During class sessions we will discuss remaining questions and problems.

References: Much of the course will follow a draft of a book by Jeremy Siek and Evan Chang. A PDF of the book is available on the course web page.

- Compilers: Principles, Techniques, and Tools, 2nd edition by Aho, Lam, Sethi, and Ullman, 2006
- Modern Compiler Implementation in Java, 2nd edition, Andrew W. Appel and Jens Palsberg, 2002

Evaluation: There will be several small assignments, including development (coding and testing), writing documentation, and preparing and giving presentations. There will also be several larger assignments, centered around developing a simple web application framework and using that framework to develop web applications. You will also be asked to develop applications on web frameworks other than your own.

In addition, students will do exercises in class and are expected to participate in classroom discussions and to make presentations to the class on aspects of web programming.

Grades will be weighted as follows:

Assignments and exercises	20%
Compiler project	50%
Exam 1	15%
Exam 2	15%