

# Compiler Design

## COMP 520

McGill University, Fall 2013

### Course Details

**Time:** Monday, Wednesday, Friday, 4:35pm-5:25pm

**Place:** ENGTR 0070

**Instructor:** Matthieu Dubet

**Office:** McConnell 234

**Office hours:** Wednesday, Friday, 2:00pm to 4:00pm

**Email:** matthieu.dubet@mail.mcgill.ca

**Teaching Assistant:** Ismail Badawi

**Office:** McConnell 234

**Office hours:** TBD

**Email:** ismail.badawi@mail.mcgill.ca

**Teaching Assistant:** Vineet Kumar

**Office:** McConnell 234

**Office hours:** Tuesday, Thursday 2:30pm to 3:30pm

**Email:** vineet.kumar@mail.mcgill.ca

### Email, Website

Students are expected to monitor their McGill email account for course-related news and information. The course website is: <http://www.cs.mcgill.ca/~cs520>.

### Prerequisites

COMP 273 (Introduction to Computer Systems), COMP 302 (Programming Languages and Paradigms). COMP 330 (Theoretical Aspects of Computer Science) is also recommended. Students without COMP 330 should familiarize themselves with regular and context-free languages.

This is a programming-intensive course. The primary languages used are C and Java and familiarity with at least one of them is assumed.

### Description

This course covers modern compiler techniques and their application to both general purpose and domain specific languages. The practical aspects focus on current technologies, primarily Java and interactive web services.

A detailed syllabus is available from the course website. This is a summary of the core topics that will be covered.

- **Deterministic parsing:** LR parsers, the flex/bison and SableCC tools.

- **Semantic analysis:** abstract syntax trees, symbol tables, type checking, resource allocation.
- **Virtual machines and run-time environments:** stacks, heaps, objects.
- **Code generation:** resources, templates, optimizations.
- **Surveys on:** garbage collection, native code generation, static analysis.

## Textbook

A course pack is available from the McGill bookstore. It consists of chapters from *Compiler Construction* by Kenneth C. Loudon and *Modern Compiler Implementation in C* by Andrew W. Appel.

The course web page also contains links to various online readings.

## Evaluation

**Assignments and project:** 65%

**Midterm:** 10%

**Final:** 25%

Both exams are closed book. Assignment marks are allocated as follows:

5% for the JOOS deliverables

10% for the JOOS peephole optimizer

50% for the WIG compiler and report

Both project and assignments are completed by groups, meaning that 65% of the marks in this course come from group work. The contributions of each group member will be considered in grading. If there is a significant difference between contributions of different group members, the grades may be redistributed accordingly.

## Assignment and Exam Policy

All assignments (deliverables, milestones, and the project) are due at midnight. We will be using Git for all group submissions. Marks will be generously deducted for lateness. No assignment submissions will be accepted after marked assignments have been returned or after solutions have been discussed in class.

In general, work submitted for this course must represent your own efforts. Copying assignments or tests, or allowing others to copy your work, will not be tolerated. Note that introducing syntactic changes into a copied program is still considered plagiarism.

McGill University values academic integrity. Therefore all students must understand the meaning and consequences of cheating, plagiarism and other academic offences under the Code of Student Conduct and Disciplinary Procedures (see <http://www.mcgill.ca/students/srr/honest/> for more information).

In accord with McGill University's Charter of Students' Rights, students in this course have the right to submit in English or in French any written work that is to be graded.