

CS 322 Languages and Compiler Design II - Winter 2005

Instructor:

Andrew Tolmach

120-23 FAB

503-725-5492

email: apt@cs.pdx.edu

Office Hours: MW 4-5 & by appt.

Course home page: <http://www.cs.pdx.edu/~apt/cs322>

Description

CS321/322 studies, in parallel, the design and implementation of programming languages and the design and implementation of compilers. The course is centered around a substantial programming project: implementing a complete compiler for a realistic language. CS322 concentrates on code generation and runtime organization; we will assume the existence of a compiler front-end, and build a back-end for it. Completed compiler projects will produce machine code that can be run directly on the target hardware. A detailed list of lecture topics may be found below.

Prerequisites

CS321 with a grade of C or better, preferably taken Fall 04 with this instructor. CS321 and 322 form a connected pair of courses; having completed CS321 in the distant past with another instructor will not necessarily have prepared you for this version of CS322. In particular, this offering of CS322 will use Java as the project implementation language.

Substantial experience with the C, C++, or Java programming language and the Unix operating system is essential. Some experience with machine-language programming is extremely desirable.

Texts

The required textbook is

- Keith D. Cooper and Linda Torczon, *Engineering a Compiler*, Morgan Kaufmann, 2004.

(This has *not* been ordered by the PSU bookstore, since most people should already have copies from last term; you can get a copy via Amazon.com or your favorite on-line dealer.) Additional readings from other sources will be handed out in class from time to time.

The official SPARC architecture manual is available on the WEB at <http://www.sparc.com/standards/V8.pdf>. If you need more information, a good auxiliary reference text is R. P. Paul, "SPARC Architecture, Assembly Language Programming, and C," Prentice Hall, 1994 (the 2nd edition of 1999 is also fine).

You'll also definitely need a Java textbook. There are many good possibilities; here are two I suggest:

- Ken Arnold and James Gosling, *The Java Programming Language*, Addison Wesley. (This book is currently in its third edition.)
- Bruce Eckel, *Thinking in Java, 3rd ed.*, Prentice-Hall, 2002. This book is available free on the web at <http://www.mindview.net/Books/TIJ>.

Lecture notes will be available electronically in pdf (Acrobat) format via the course home page.

Project

Over the course of CS322 you will complete the PCAT compiler you began in CS321. You will also build an interpreter for the language. A working version of the front end from CS321 will be available for you to use in place of your own, if you prefer.

You are strongly encouraged – but not required – to work in teams of two on the project assignments. Team members submit a single version of each project assignment, and receive the same grade.

Your compiler must be written in Java. Substantial supporting code will be handed out by the instructor.

Exams

There will be one mid-term and a final exam. Both are *closed-book*. Exams will cover topics from lectures and readings, emphasizing material that is not directly relevant to the programming project. Not all the material is covered in the readings, so lecture attendance is important. Exams are scheduled in advance; unless prior arrangements are made, a grade of zero will be recorded for missed exams.

Homework problems may be assigned from time to time as an aid to help you study for the exams; they will not be collected or graded.

Grading

Approximately 1/2 on programming assignments and 1/4 on each of the two exams (including the final).

Programming assignment grades will be determined primarily by observing program behavior on test inputs. Computer programming being what it is, this policy means that “small” errors in your code can have a large effect on your grade. Sample test inputs and a correctly-behaving executable will be provided for you to compare your program against. It is your responsibility to apply these tests and others of your own devising before submitting your assignment. We will apply some non-public tests to your programs as well.

Computing Facilities

The project will be implemented in Java and SPARC assembly language. For most of the assignments, you will need access to a SPARC processor, such as those found on the CS department’s network. When feasible, you may develop your project solutions on any machine you like, but the final version of *all* assignments must work using the CS department’s Sparc machines, using Java SDK version 1.4.2 and the GNU SPARC assembler (`/pkg/gnu/bin/as`).

You should have been given an account on the Solaris machines automatically by virtue of registering for this course; to obtain your password, take your PSU id to one of the CAT front desks (e.g. FAB 135-01). Once your account is setup, perform “`addpkg`” commands to add `jdk1.4.2` and `gnu` to your environment. Documentation for this version of the Java can be found on the web at <http://java.sun.com/j2se/1.4.2/docs/index.html>.

Files associated with assignments will be made available for download via the course web page and/or placed in `/u/cs322-01` on the Solaris network.

Submitting Programs

Programs should be submitted by email to `cs322-01@cs.pdx.edu`. Details on how to submit programs will be provided with each assignment. Do not send mail other than program submissions to `cs322-01`, as it is not regularly read.

Mailing List

Important information will be distributed throughout the term via a mailing list called `cs322list`. To subscribe to the list, visit the course home page and follow the directions from there. Please mail questions to the instructor directly (at `apt`) rather than to this list; the instructor will copy mail of general interest to the list. The list is archived so you can consult old messages.

Schedule

All dates, including assignment deadlines, are tentative.

Date	Asgn	Reading	Topics
Jan	3		introduction; synopsis of backend tasks
	5	1	interpreters; review of attribute grammars
	10	CT 6.2,6.3.1	environments; procedure activations and scope
	12	2	skim Sparc 2-5,A,D
	17		Martin Luther King, Jr. Holiday (NO CLASS)
	19		semantics
	24	1*	language elements: expressions and statements
	26	CT 5.1-4	intermediate representations
	31	CT 7.1-7	IR generation: booleans; backpatching
Feb	2	2* 3	IR generation: statements
	7		Midterm Exam (in class)
	9	CT 6.3.2,6.5-6,7.9	compiled runtime organization
	14	CT 6.4	parameter passing mechanisms
	16		procedures as parameters; functional programming
	21	4	CT 6.3.3,7.10
	23	3*	CT 11.1-3
	28		CT 8
Mar	2		CT 13.1-4,13.5.1-4
	7		CT 12.1-3
	9	4*	CT 6.7
	16		Final Exam (Wednesday starting at 12:30 pm)

Readings column key: CT = Cooper and Torczon textbook; Sparc = Sparc architecture manual.

Additional readings may be assigned from time to time. Assigned selections should be read *before* the associated date.

Assignments

Programming assignments, listed below, are distributed on the dates the assignment number appears in the schedule above and are due when the starred number appears.

- 1 interpreter
- 2 SPARC machine code exercise
- 3 preliminary code generation
- 4 final code generation

Rules

Programs are due *by 1:30 p.m.* on the specified due date, i.e., just *before* class. Late programs are *not* accepted except in extraordinary circumstances, and then preferably by prior arrangement. The project deadlines are there to help you, by forcing you to keep up during the term.

All programming assignments must represent the work of your explicitly identified two-person team. It is permissible to discuss the assignment with other students, but you must develop the solutions yourselves. *Do not, under any circumstances, copy another person's program and submit it as your own.* Writing code for use by another or using another's code in any form (even with their permission) will be considered cheating. In particular, cheating will result in an automatic zero grade for that piece of work, and the initiation of disciplinary action at the departmental and University level.

If you are a student with a disability in need of academic accommodations, you should register with Disability Services for Students and notify the instructor immediately to arrange for support services.