CS 457/557 Functional Programming - Spring 1997

Instructor:
Andrew Tolmach
PCAT 128A
725-5492
email: `apt@cs.pdx.edu`
Office Hours: TuTh 3:30-4:30 & by appt.

Course home page: `http://www.cs.pdx.edu/~apt/cs457/cs457.html`

Description

This course provides a gentle introduction to the ideas and techniques of functional programming, using the Standard ML (SML) programming language. Functional languages are great for both rapid prototyping and serious software engineering. The key idea is to program declaratively using function definitions rather than operationally using assignments. SML's features include higher-order functions, strong typing, polymorphism, user-defined algebraic data types, automatic storage management, and powerful abstraction and modularization facilities. The course will concentrate on developing practical programming skills in SML, with a brief look at the theoretical underpinnings.

Prerequisites: CS202, CS252.

Textbooks

Chris Reade, "Elements of Functional Programming," Addison Wesley, 1989.
Jeff Ullman, "Elements of ML Programming," Prentice-Hall, 1994.

Lecture notes will be available electronically in postscript format via the course home page.

Requirements

There will be two in-class, open-book exams, each worth about 25%, and a programming project, worth about 50%. There is **no** final exam. There is also **no** graded homework, although problems will be assigned regularly to help you prepare for the exams.

The project requires you to write a substantial program in SML; the application area is your choice, subject to instructor's approval. Good application areas include graphics and language processing. Projects may be done individually or in teams of up to three. A firm project specification is due on May 13; projects themselves are due on June 3, and selected projects will be demoed in class on June 5.

Computing Facilities

To do the exercises and the project, you can use either the Standard ML of New Jersey (SML/NJ) compiler (version 0.93) or the Moscow ML interpreter (version 1.41). Both are available on the CS department's Sparc network; if you do not have an account on this network, you should obtain one immediately by following the instructions on The Form, available in the CS department office. Both implementations support an interactive "read-eval-print" loop interface, into which you can type new code or load existing files. To use SML/NJ, do "`addpkg sml`"; the executable is called `sml`. To use Moscow ML, do "`addpkg mosml`"; the executable is called `mosml`.

Moscow ML, which uses far less memory than SML/NJ, is also available to run on PCs (under DOS or unix variants) and Macs. You are encouraged to try this at home! See the course home page for pointers to the software distributions.

Mailing List

Important information will be distributed throughout the term via a Majordomo list called `cs457list`. You can subscribe to this list by sending a mail message to `majordomo@cs.pdx.edu` with no subject line and a message body consisting of the single line

```
subscribe cs457list
```

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.

Tentative Schedule

Apr 1	Introduction
3	Functions and types
8	Recursion
10	Lists
15	Higher-order functions
17	Polymorphism
22	Concrete types
24	Review
29	**Exam I**: Reade 1-5; Ullman 1-7,10-13,19,21
May 1	IO and Exceptions
3	References and Arrays
8	Graphics with ML/Tk
13	Abstraction Mechanisms; **Project Specifications due**
15	Modules
20	Pragmatics
22	**Exam II**: Reade 7,9; Ullman 8-9,14-17,20
27	Lazy Evaluation (Reade Ch. 8)
29	Implementation of ML
June 3	Wrap up; **Projects due**
5	Project Demos

Note: There is **no** final exam.