CSE 341 – Overview, Autumn 1999

Personnel

Instructor: *Jason Hartline*, `hartline@cs.washington.edu`, (206) 616-2405
Office Hours: Sieg 226C, Mon. Fri. 2:30-3:30 or by appointment.

Teaching Assistant: *Douglas Low*, `douglas@cs.washington.edu`
Office Hours: Sieg 226A, Wed. 2:30-3:30; Thurs. 1:30-2:30.

Class Meetings

Lectures	MWF 1:30-2:20	MLR 301
Section AA	Th 8:30-9:20	EEB 216
Section AB	Th 9:30-10:20	MEB 248

Objectives

Our objective is to learn fundamental programming language concepts. We approach this by acquiring practical experience with a set of four very different programming languages – Scheme, Java, Miranda, and CLP(R). Following the study of the individual languages, we'll finish up by a comparative discussion of programming language concepts in these and other languages.

Prerequisite: CSE 143

Text Books

There are no required text books; however, the following are recommended. They are on reserve in the Engineering Library.

- Abelson and Sussman, *Structure and Interpretation of Computer Programs.*

  This is a reference for Scheme – it's actually the introductory programming text used at MIT and is a great book, not just for Scheme but for computer science concepts generally.

- Timothy Budd, *Understanding Object-Oriented Programming with Java.*

- Simon Thompson, *Miranda: The Craft of Functional Programming.*

You can also find two other books in the Engineering Library:

- Ravi Sethi, *Programming Languages: Concepts & Constructs*, 2nd Ed.

  This is a good reference for many of the concepts we will be discussing in the class. Unfortunately it does not present these concepts with the languages that we teach.

- Marriott and Stuckey, *Programming with Constraints.*

Email List

The course email list is `cse341@cs`. It is a majordomo mailing list. Subscribe in the usual way. Send email to `majordomo@cs` with the text, `subscribe cse341`, in the body of the message. You are responsible for announcements made to this list.

Computing Resources

The supported course software is on the instructional Linux machines; `tahiti`, `sumatra`, `ceylon`, and `fiji`. You are welcome to work on your homework on your own machines using interpreters/compilers that are not supported by the course; however, it is your responsibility to make sure that your homework works on the course supported software also.

Scheme We we will be using the *Guile* Scheme interpreter. You are encouraged to work from home. Information on *Guile* is available from `http://www.gnu.org/software/guile/guile.html`. You should also feel free to work on homework with alternative scheme interpreters such as *Dr. Scheme* which is available from `http://www.cs.rice.edu/CS/PLT/packages/drscheme/`.

Java We will be using Java 2.0 with an implementation of Sun's JDK 2.0 that comes from `http://www.blackdown.org/`. There are also versions of Java 2.0 installed on the NT machines in Sieg 232.

Miranda The Miranda interpreter, `mira`, is commercial software.

CLP(R) The CLP(R) interpreter is `clpr`.

Grading

The course will consist of about eight assignments, roughly two for each language. One of these assignments will be a warmup and the other will be a more substantial exercise. There will be a midterm and a final.

homework	50%
midterm	20%
final	30%

Individual grades may vary slightly, based on effort, contribution to class and section, etc.

Cheating

Students are expected to do the assignments on their own, except for assignments explicitly labelled as group assignments. Any cases of cheating that we discover will be sent to the College disciplinary committee.

However, we also want to be clear on what is legitimate collaboration – please help each other out in this class in appropriate ways! It is OK to help other students debug their programs, and to discuss general approaches to solving problems. After having such a discussion, though, you should go do something else for at least half an hour, for example watch an inane TV show, before independently working on your solution (This is sometimes called the Gilligan's Island rule). It is not OK to copy someone else's code or homework solution.

Exams must of course be done on your own.

Late Assignments

Assignments will typically be due on Monday. Unless otherwise stated, they should be turned in during class, but will also be accepted until 5:00 in 226C. Under certain circumstances extensions may be granted; however, it is your responsibility to let us know before hand (at least the lecture or section before the assignment is due – typically Friday). For more details see *Submission Guidelines.*