CS550 – Programming Languages and Systems
(Programming Paradigms, Semantics)

Presentation and Syllabus

Manuel Hermenegildo
herme@unm.edu
Room CS 340

Departments of Computer Science and
Electrical and Computer Engineering
U. of New Mexico
Albuquerque, NM – USA

Fall 2006

Note: Course slides (including the information on systems, etc. in this set) may change during the semester. If you are looking at a printed version it is always advisable to check the on-line version for changes at http://www.unm.edu/~herme/courses/CS550

Course Description

CS550 – Programming Languages and Systems (Programming Paradigms, Semantics)

A course on current trends in design and philosophy of programming languages and systems. In particular, the course covers topics in programming paradigms (logic programming, functional programming, constraint programming) and semantics. Concepts, theoretical foundations, and programming practice will be addressed.

Course Description – Prerequisites / Emphasis

- **Prerequisites:**
  - Proficiency in imperative and object-oriented programming is assumed. Also, some familiarity with functional programming.
  - Some knowledge of logic programming and/or proficiency in functional programming are useful but not strictly required.
  - CS451 prerequisite.
- **Emphasis note:**

  Because familiarity with imperative and object-oriented programming (and some familiarity with functional programming) are assumed, the course concentrates first on reviewing other programming paradigms that the student may be less familiar with (e.g., logic, functional-logic, and constraint programming), including their combinations.
- The presentation (topics/speed) will be adapted if possible to student backgrounds.

General Information

Instructor:	Manuel Hermenegildo
Office:	CS 340
Class time:	Tuesday and Thursday 14:00–15:15
Class venue:	ME 208
Office hours:	Tuesday and Thursday 15:30–16:30 p.m., or by appointment and emai .
email:	herme@unm.edu

- Information on the course available on-line via WWW
  (for reading on-line or printing out)
  - ◇ Transparencies for all lectures (including these) in several formats:
    pdf, *postscript*, and *html*.
  - ◇ Access to mailing list archive.

  Follow the CS550 link from: http://www.unm.edu/~herme/
- Standard "mailman"-based course mailing list:
  - ◇ Sign up at: https://www.cs.unm.edu/cgi-bin/mailman/listinfo/cs550
  - ◇ Send messages to: cs550@cs.unm.edu

Assignments and grading

- (40%) Homework assignments.

  Expected topics:

  1. Logic Programming.
  2. Combined Functional Programming and LP.
  3. Constraint programming.
  4. Semantics

- (35%) An individual term project, including in-class presentation.
  - Topic will be of your choice,
  - Watch out in class for ideas.
  - You need to have the topic proposed and approved before mid-term (ideally much earlier).

- (25%) Short midterm and final, in-class, individual tests.

Assignments and grading (Contd.)

- Homework and programming assignment hand-in:

  Homework assignments are due at midnight of the due date assigned. Late homework will be given quickly diminishing credit as the deadline passes. Hand-in mechanisms (usually electronic) will be specified with each assignment.

- Academic honesty:

  You are welcome to *discuss* homework with your classmates and the instructors. You must *do* and *write up* your own solution. You cannot look at or copy someone else's solution and you also cannot provide your solution to another student. You are responsible for protecting your homework from unauthorized access. You should ask the instructor promptly if you lack the background to solve a question or find it too difficult.

Other

- Class attendance is mandatory.
- UNM statement of compliance with ADA:

  Qualified students with disabilities needing appropriate academic adjustments should contact the instructor as soon as possible to ensure their needs are met in a timely manner. Handouts are available in alternative accessible formats upon request.

Course Outline

- Introduction and Motivation. Programming paradigms.
- Logic Programming:
  - Pure logic programming, Prolog, and other LP systems.
  - Operational semantics and fixpoint semantics.
- Functional+logic Programming, higher order.
- Constraint Programming:
  - Intro, CLP(R), CLP(Q), finite domains.
  - Operational, model, and fixpoint semantics.
- Semantics wrapup:
  - Main types of semantics. Operational semantics. Denotational semantics.
  - Axiomatic semantics.
  - Introduction to program verification.

Programming Environment

- Programming environment used in the course: 
  - Multi-paradigm programming environment supporting:
    - Logic programming
    - Functional programming
    - Constraint programming
    - (A form of) Imperative, object-oriented programming

    (allows following the course with a single system)
  - Installing/Using Ciao:
    - Freely available from: http://www.ciaohome.org
    - You can easily install it in your machine (Linux, Unix, Windows, & Mac).
    - If needed, it can be installed on the CS Linux machines.
  - Use of the (`emacs`-based) programming environment highly recommended (syntax highlighting, auto-documenter, multiple debuggers, assertions, etc.).

Full info at:

http://www.unm.edu/~herme

http://www.unm.edu/~herme/courses/CS550