

University of Illinois at Chicago  
Fall 2005

## CS 476 / MCS 415 — Programming Language Design Course Syllabus

**Room:** A5 LC

**Time:** MWF 12 - 12:50

**URL:** via Blackboard, under Engineering: CS - Computer Science: Fall 2005 CS476

### Staff

**Instructor:** Barbara Di Eugenio

**Office:** 916 SEO

**Phone:** 6-7566

**E-mail:** bdi Eugen@uic.edu

**Office Hours:** Wed 3-4:30

**Teaching Assistant:** Davide Fossati

**TA's email:** dfossal@uic.edu

**TA's Office Hrs:** TR 3:30-4:30, 938 SEO

### Course Objectives

The aim of this course is to provide students with the tools necessary for the critical evaluation of existing and future programming languages. We will do so by examining the process design and features of three of the four main types of programming languages: imperative (C, Fortran, C++, Java); functional (Common Lisp, Scheme, ML, Haskell); logic (Prolog). In addition, we will touch on some object-oriented programming features of C++ and Java (object-oriented programming is not a focus of this class). Students will be exposed to two programming paradigms that are usually not taught in other classes: functional (mainly via Common Lisp) and logic (via Prolog).

### Textbooks

**Required:** Robert W. Sebesta. Concepts of Programming Languages (7th edition). Addison Wesley, 2006.

**Note for students registered under MCS415:** In the bookstore, the book is available under CS476, not MCS415.

**Highly Recommended** book on Lisp: Paul Graham. ANSI Common Lisp. Prentice Hall, 1996.

I will also distribute / refer you to other papers as necessary.

An on-line tutorial on Common Lisp is *Lisp Primer*, by Collin Allen and Maneesh Dhagat: <http://mypage.iu.edu/~colallen/lp/>.

An on-line tutorial on Prolog can be found at <http://computing.unn.ac.uk/staff/cgpb4/prologbook/>

### Prerequisites

CS 340 (for CS students), MCS 360 (for MCS students)

## Notes

- I use email and the web page message board a lot to communicate with the whole class. Please check your email frequently, especially around deadlines (homeworks and exams). You should use the web page message board to post any questions of general interest, such as clarification on the homework etc. I won't answer general questions by email but on the board.
- The web page will contain all materials relevant to the class, syllabus, assignments, lecture notes, additional papers, etc. You can also see your own grades.

## Tentative Schedule

Dates	Topic	Readings
Week 1	Introduction	Ch. 1-2
Week 2	Syntax, Semantics	Ch. 3
Week 3	Names, Scope	Ch. 5
(9/5)	<b>Labor Day, no class</b>	
Week 4	Data Types, Expressions	Ch. 6-7
Week 5-7	Functional Programming: Common Lisp	Ch. 15, Lisp Primer
Week 8	Functional Programming: Haskell	Ch. 15
Week 9-10	Logic programming: Prolog	Ch.16, Prolog materials
Week 11	Control Structures	Ch. 8
Week 12-13	Subprograms	Ch. 9-10
Week 14	Concurrency, Exception Handling	Ch. 13-14
(11/24)	<b>Thanksgiving, no class</b>	
Week 15	Catch up, Review, etc ...	

## Important Dates

Note: homework deadlines are tentative. Paper and pencil homeworks will be submitted at the beginning of class, programming homeworks will be submitted electronically by midnight (i.e., 11:59pm) on the day they are due.

Date	Event
9/16 (Fr)	Homework 1 due
9/28 (Wed)	<b>Midterm 1</b>
10/17 (Mon)	Homework 2 due
11/4 (Fr)	<b>Midterm 2</b>
11/9 (Wed)	Homework 3 due
11/30 (Wed)	Homework 4 due
Finals week (12/5-9)	<b>Final</b>

## Grading Criteria

- **4 Homework Assignments** (40% total): Each homework will be worth between 8% and 12% of the grade.
- **3 Exams:** 2 midterms (each worth 15%), 1 final (30%).

**Important Note:** To pass the class you must get at least 60% of the total available points for the three exams.

Letter grades will be decided **only at the end**. However, the following guidelines will be adhered to:

Overall Score (undergraduate)	Overall Score (graduate)	Letter grade
88%	92%	A
78%	82%	B
68%	72%	C

## Policies on homeworks and exams

### General Policies

1. Late homeworks will not be accepted in any case, unless there is a **documented** personal emergency. Arrangements must be made with the instructor as soon as possible after the emergency arises, preferably before the homework due date.  
**Advice:** If for whatever reason you don't manage to finish an assignment, hand in what you have. Partial credit may be given at the grader's discretion.
2. Statute of Limitations: **Two weeks!** No grading questions or complaints — **no matter how justified** — will be listened to two weeks after the item in question has been returned.

### Homeworks

There will be 4 homeworks. Homeworks will be a mixture of pen and pencil and programming assignments. Homeworks will be handed either in class, or via the facility available under the web page. More details will be available later.

### Exams

1. The two midterms will be given during class time; consequently, **no make-ups** will be given.
2. Exams will be closed-book.
3. The final is cumulative, with more emphasis on the last part of the class.

## Policy on Academic Integrity

**Academic dishonesty will not be tolerated.** Please see the CS department policy below on the topic; this policy specifies penalties for violations.

What is academic dishonesty? To hand in any work which is not 100% the student's creation, unless you are explicitly allowed to do so. Thus:

1. **Exams.** All work on all exams must be individually performed.
2. **Homeworks:** no student may give any other student any portion of their solutions or code, through any means. Students are not allowed to help each other debug the code, or to show each other any portions of code or homework.

**Important Note:** every semester somebody is caught red-handed and as a consequence fails the class. Isn't it better to get a B or a C than an E?

### CS department policy on academic dishonesty

The CS Department will not tolerate cheating by its students. The MINIMUM penalty for any student found cheating will be to receive an E for the course and to have the event recorded in a department and/or College record. The maximum penalty will be expulsion from the University.

We intend to devote more effort than in the past to detecting and punishing cheating. Cheating includes all the following, though this is not a complete list:

- Copying or any other form of getting or giving assistance from another student during any test, quiz, exam, midterm, etc.
- Plagiarism—turning in writing that is copied from some other source.
- Obtaining solutions to homework by posting to the Internet for assistance, purchasing assistance, obtaining copies of solutions manuals for instructors, and obtaining copies of previous year's homework solutions.
- Computer programs: Any time you look at another student's code, it is cheating. (Exception: If you are EXPLICITLY told that you may do so by the instructor, for instance, in working on a large group project in the 400-level software engineering course.)

For computer programs, if for some reason we cannot determine who copied from whom, we may, at our discretion, give failing grades to both students.

It is the responsibility of all engineering and computer science professionals to safeguard their company's "trade secrets." An employee who allows trade secrets to be obtained by competitors will almost certainly be fired. So, YOU are responsible for making sure that your Unix directories have permissions set so that only you can read your files, for being sure to log out at the end of working in the computer lab, etc.