

# CompSci 431: Programming Languages Concepts—Fall 2017

John Boyland

Instructor	John Boyland
Office	925 EMS, 229-6986
Office Hours	R 10:00–12:00pm, F 8:00–9:00am
Course Homepage	<a href="http://www.cs.uwm.edu/classes/cs431/">http://www.cs.uwm.edu/classes/cs431/</a>
Twitter Feed	@boylanduwm
Questions?	boyland@uwm.edu

## 1 Introduction

CompSci 431 introduces a variety of programming languages for those who already know at least one (we expect Java). The course briefly covers ML, JavaScript and Prolog, but also gives a broad overview to the field of programming languages. In this section of the course, we will use `git` to handle assignments.

## 2 Outcomes

1. Students will be able to understand and write context-free grammars using BNF/EBNF notations and syntax diagrams
2. Students will be able to understand core programming language concepts that include types, polymorphism, scopes, memory allocation of variables, memory management, parameter passing styles, and formal semantics.
3. Students will be able to understand and use the concepts of
  - (a) functional programming (pattern matching, type inference, parametric polymorphism, nested functions, higher-order functions, data and type constructors),
  - (b) object-oriented programming (inheritance, encapsulation, subtyping polymorphism, dynamic dispatch, and exception handling)
  - (c) logical programming (resolution and unification).
4. Students will be able to write simple programs using the programming languages ML, JavaScript, and Prolog.

## 3 Requirements

Students must have received a ‘C’ or better in CompSci 351. In particular, this class requires that student be able to use and implement

- standard data structures (dynamic arrays, linked lists, binary search trees) and
- standard abstract data types (lists, tables, graphs, stacks, queues).

## 4 Texts

The required textbook for the course is

Adam Brooks Webber. *Modern Programming Languages: A practical introduction*. Franklin, Beedle and Associates, Inc., 2nd ed., 2011.

Additionally there will be readings from the open book JavaScript Guide available on the Mozilla Developer Network.

## 5 Grading

The grade for the course will be computed from the following parts:

### 40% Exams

There will be one midterm and one final exam, each worth 20% of the grade. The midterm will be given over two days.

### 60% Homework

There will be a homework every week (13 in all), partly programming and partly text. Each homework is due at the start of lecture. Late homework will not be accepted. Each homework is worth 5% of the grade. The lowest homework grade will be dropped.

Exceptions to these rules can be made only in extraordinary circumstances. Advance notice of a need for an exception should be given whenever possible.

### 5.1 Criteria

Programs and written assignments will be graded for correctness, suitability, style, clarity and practicality. Although we will provide solutions to some assignments, there are usually a wide variety of correct answers to any particular assignment.

### 5.2 Academic Honesty

Unless an assignment explicitly says otherwise, all graded assignments must be *your own work* (your own words), but you may work with other people as long as you list their names prominently on the first page of the assignment, and/or in a comment at the top of the assignment. For this course, verbal communication and collaboration using non-code text or hand-written code is permitted, as long as it is properly documented. Documentation must also be made for help from anyone not in the course, such as a tutor, friend, or relative, and for information off the Web.

You may not make copies for other people of assignments, including through email, texting, scanning, photography, or any other copying technique, except where specifically permitted in an assignment. Whether or not you have permission of the other, submitting someone else's work as your own is plagiarism, a serious instance of academic misconduct. Everyone is responsible for learning the material themselves. Some of the assignments may be graded in person, especially in cases where the individual contribution to the assignment is not clear. If you are graded in person, you will be expected to demonstrate that you have mastered techniques used in the material you submitted.

In this course, you will be running code written by course staff. None of these programs are "Trojan horses": none will ever attempt to perform covert tasks that invade privacy or destroy data. Similarly, you may not submit Trojan horses as regular assignments. Any such action will be dealt with severely.

### 5.3 Course Grades

At the end of the course, the numeric grade will be converted into a letter grade according to the following scale:

Letter Grade	A	A-	B+	B	B-	C+	C	C-	D+	D	D-	F
Minimum Score	92	90	87	82	80	77	72	70	67	62	60	0

There is no curve, but the instructor reserves the right to increase a grade if he believes it would not reflect the student's mastery of the material. A grade can only be decreased for academic dishonesty.

## 6 Workload

This course meets weekly for 45 contact hours. You also have required reading of about 500 pages, which we estimate will take approximately 50 hours. There are thirteen required homework assignments, each of which will take 10 or more hours. This yields a total estimated workload of 225 hours.

## 7 Schedule

Week starting	Topic	Reading (Default MPL)
9/6	Introduction; Syntax	Ch. 1, 2
9/11	Abstract Syntax; Language Systems	Ch. 3–4
9/18	Introduction to ML; Type	Ch. 5–6
9/25	ML Patterns and Nesting; Polymorphism	Ch. 7–8
10/2	Higher Order; Scope	Ch. 9–10
10/9	ML Datatypes; Activation Records	Ch. 11–12
10/16	Introduction to JavaScript; Memory Management	JSG 1–8, MPL Ch. 14
10/23	<b>Midterm</b> (both days)	JSG 1–8, MPL Ch. 1–12,14
10/30	JS Objects	JSG 9–12, MPL Ch. 16
11/6	JS Iterators and Generators; Exceptions	JSG 13, MPL Ch. 17
11/13	Parameters; Introduction to Prolog	Ch. 18–19
11/20	Procedural Prolog; Cost Models	Ch. 20–21
11/29	Arithmetic in Prolog	Ch. 22
12/7	Semantics	Ch. 23
12/14	History of Programming Languages	Ch. 24
12/22	7:30am: <b>Final Examination</b>	

## 8 Notes

- Resources (including copies of all slides) are available on the web.
- AFS is available at <http://www.openafs.org>; free downloads are available.
- If you will be needing any accomodation in this course for any reason, please contact the instructor. Please also be aware of the standard University policies: <http://www.uwm.edu/Dept/SecU/SyllabusLinks.pdf>