

CMSCXXXX: Types and Programming Languages

Course Description

This course provides an introduction to type systems and the basic theory of programming languages. Part I of the course will focus on the Simply Typed Lambda Calculus (STLC), and then extend that language with features such as algebraic data types (ADTs), references, and exceptions. For each extension, we will revise a proof of type safety. Part II of the course will focus on ["fancy"](#) type extensions to the STLC including parametric polymorphism, recursive types, and subtyping.

Course Details

- **Course:** CMSCXXXX
- **Prerequisites:** CMSC330
- **Credits:** 1
- **Seats:** TBD
- **Lecture Time:** TBD
- **Location:** TBD
- **Semester:** Spring 2020
- **Textbook:** [Types and Programming Languages](#) (optional, readings will be provided)
- **Course Facilitator(s):** Ian Sweet
- **Faculty Advisor:** Michael Hicks

Topics Covered

Syllabus may be subject to minor changes, but drastic revisions will require input of students/facilitators, and those involved will be notified immediately

- Part I: Simple Types
 - Simply Typed Lambda Calculus (Ch. 9)
 - Types (Ch. 9.1)
 - The Typing Relation (Ch. 9.2)

- Properties of Typing (Ch. 9.3)
- Simple Extensions (Ch. 11)
 - Base Types (Ch. 11.1 - 11.2)
 - Derived Forms: Sequencing, Wildcards (Ch. 11.3)
 - Type Ascription (Ch. 11.4)
 - Let-Binding (Ch. 11.5)
 - Pairs (Ch. 11.6 - 11.8)
 - Sums (Ch. 11.9 - 11.10)
 - General Recursion (Ch. 11.11)
- References (Ch. 13)
 - Overview (Ch. 13.1)
 - Typing (Ch. 13.2)
 - Evaluation (Ch. 13.3)
 - Store Typings (Ch. 13.4)
 - Type Safety (Ch. 13.5)
- Part II: Fancy Types
 - Subtyping (Ch. 15)
 - Subsumption (Ch. 15.1)
 - The Subtype Relation (Ch. 15.2)
 - Properties of Subtyping and Typing (Ch. 15.3)
 - The Top and Bottom Types (Ch. 15.4)
 - Subtyping and Other Features (Ch. 15.5)
 - Java's covariant arrays :(
 - Intersection and Union Types (Ch. 15.7)
 - Recursive Types (Ch. 20)
 - Examples (Ch. 20.1)
 - Formalities (Iso- vs. Equi-recursive) (Ch. 20.2)
 - Polymorphism (Ch. 22 - 24)
 - Type Reconstruction (Inference) (Ch. 22)
 - Universal Types (Ch. 23)
 - Existential Types (Ch. 24)
 - Parametricity
 - Termination: Tait's Method (Ch. 12, PFPL supplement)
 - Parametricity: Girard's Method (PFPL supplement, Derek's notes)

Schedule

Week	Topic	Assignment
1 (01/27)	Simply Typed Lambda Calculus	Reading: Ch. 9.0 - 9.3 Due: None Assigned: PS1
2 (02/03)	Simple Extensions	Reading: Ch. 11.0 - 11.11 (whatever interests you) Due: PS1 Assigned: PS2
3 (02/10)	References	Reading: Ch. 13.0 - 13.5 Due: PS2 Assigned: PS3
4 (02/17)	Subtyping: Subsumption & Type Safety	Reading: Ch. 15.0 - 15.3 Due: PS3 Assigned: PS4
5 (02/24)	Subtyping: Top, Bottom, Java, Union + Intersection Types	Reading: Ch. 15.4 - 15.5, 15.7 Due: PS4 Assigned: PS5
6 (03/02)	Recursive Types	Reading: Ch. 20.0 - 20.2 Due: PS5 Assigned: PS6
7 (03/09)	MIDTERM	Reading: None Due: PS6 Assigned: Nothing
8 (03/16)	SPRING BREAK	Reading: None Due: Nothing Assigned: Nothing

9 (03/23)	Polymorphism: Type Reconstruction (Inference)	Reading: Ch. 22.0 - 22.7 Due: Nothing Assigned: PS7
10 (03/30)	Polymorphism: Universal Types	Reading: 23.0 - 23.10 Due: PS7 Assigned: PS8
11 (04/06)	Polymorphism: Existential Types	Reading: 24.0 - 24.3 Due: PS8 Assigned: PS9
12 (04/13)	Termination: Tait's Method	Reading: Ch. 12.1, PFPL Supplement , Semantics of Types (Derek's course notes) Due: PS9 Assigned: PS10
13 (04/20)	Parametricity: Girard's Method	Reading: PFPL Supplement , Semantics of Types Due: PS10 Assigned: PS11
14 (04/27)	BUFFER: if/when we fall behind	Reading: Due: PS11 Assigned: Practice Problems for Final
15 (05/04)	FINAL	

Grading

Grades will be maintained on **grades.cs**. You will be responsible for all material discussed in lecture as well as other standard means of communication (Piazza, email announcements, etc.), including but not limited to deadlines, policies, assignment changes, etc.

Any request for reconsideration of any grading on coursework must be submitted within one week of when it is returned. No requests will be considered afterwards.

Your final course grade will be determined according to the following percentages:

Percentage	Title	Description
40%	Problem Sets	Exercises from the reading, with option of OCaml programming.
30%	Midterm	The midterm will be on topics from weeks 1-6, and will consist of multiple choice, short answer, fill in the blank, etc.
30%	Final Exam	The final exam will cover all the topics discussed during the semester and will consist of multiple choice, short answer, fill in the blank, etc.

Communicating with course staff

Other means of communication have not been chosen as of now.

Interaction beyond the classroom is encouraged, but should be limited to important or more urgent issues. Topics that need not be addressed immediately can wait till class time.

Instructor(s) Name(s) and Email(s):

- Michael Hicks: mwh@cs.umd.edu

Facilitator(s) Name(s) and Email(s):

- Ian Sweet: ins@cs.umd.edu

Excused Absence and Academic Accommodations

See the section titled "Attendance, Absences, or Missed Assignments" available at [Course Related Policies](#).

Disability Support Accommodations

See the section titled "Accessibility" available at [Course Related Policies](#).

Academic Integrity

Note that academic dishonesty includes not only cheating, fabrication, and plagiarism, but also includes helping other students commit acts of academic dishonesty by allowing them to obtain copies of your work. In short, all submitted work must be your own. Cases of academic dishonesty will be pursued to the fullest extent possible as stipulated by the [Office of Student Conduct](#). It is very important for you to be aware of the consequences of cheating, fabrication, facilitation, and plagiarism. For more information on the Code of Academic Integrity or the Student Honor Council, please visit <http://www.shc.umd.edu>.

Course Evaluations

If you have a suggestion for improving this class, don't hesitate to tell the instructor or TAs during the semester. At the end of the semester, please don't forget to provide your feedback using the campus-wide CourseEvalUM system. Your comments will help make this class better.

Thanks to the CS professors at the University of Maryland, College Park for the basic syllabus outline.