CMSC245 Syllabus — Fall 2009

Part I: Languages and Language Implementations (3 lectures)
Textbook Chapter 1, Chapter 10.3 “Review/Overview of Scheme”, Chapter 5.1-5.3
• Paradigms, Languages, “Core Language System” (interpreters, compilers, etc)
• Expressing algorithms in Scheme, C++, and Assembly language (contrast with Python)
• Expressing data structures in Scheme, and C++ (contrast with Python)
• (Occasional Discussions of History and of Language Design Goals)
Labs 0 and 1: Programming (and data structures) in Scheme, C++, and HERA

Part II: Lexical and Syntactic Analysis, Parse Trees and Abstract Syntax Trees (4 lectures)
Chapter 2.1, 2.3 through end of 2.3.1 plus pages 77-80, skim 2.2
• Lexical scanners and Parsers
• Derivations, Parse Trees, and Abstract Syntax Trees
• Ambiguity, Associativity, and Precedence
• Parsers and Classes of Grammars, Predictive Parsing (First and Follow sets)
Review Questions: 2.1, 2.7, 2.9b, 2.13, 2.15
Lab 2: Predictive parser/translator

Part III: Scoping, Symbol Tables, Translation of Simple Expressions and Variables (4 lectures)
Chapter 3 through the end of 3.4, Chapter 6.1-6.5
• Translation of Arithmetic Expressions
• Variables, Variable Attributes, Scope, and Lifetime
• Symbol Tables and Symbol Table Data Structures
• Environments, Storage Allocation, and access to Local, Global, “Semi-local” Variables
• Side Effects and Order of Evaluation, Defining Language Semantics
Review Questions: 3.2, 3.5, 3.9, 3.13, 3.16, 3.18 (skip shallow-binding questions in 2009)
Lab 3: Scoping rules
Lab 4: Symbol tables

Part IV: Data Types, Data Abstraction (3 lectures)
Chapter 7.1-7.4, 7.10
• Types: Built-in and Constructed; Concrete and Abstract
• Recursive Types (Functions, Records, Pointers)
• Type Equivalence, Type Checking, Type Conversion
• Data Abstraction
• Overloading, Polymorphism, and Type Parameters
Review Questions: 7.1, 7.19, 7.20, 7.21, 7.23

Midterm Exam (Parts I - IV) five days starting Friday of the 7th week of class.
Part V: Parameter Passing and Storage Management (3 lectures?)
Chapter 7.7, 7.8, 8-8.5
- Parameter Passing and Result Returning Mechanisms
- Garbage Collection
- Exception Handling
Review Questions: 8.3, 8.4, 8.8, 8.9, 8.10, 8.13, 8.14, 9.19
Lab 5: Reference Counting

Part VI: Pure Functional Programming (3 lectures?)
Chapter 10
- Review of Basic Functional Programming Techniques (including tail recursion)
- “Currying”, Continuation-Passing-Style, other interesting parts of the Functional Paradigm
- Higher Order Functions and Implementation Thereof (closures)
- Defining Evaluation in Pure Functional Languages
- Mathematics of Functional Programming (time permitting)
Review Questions: 10.3, 10.7, 10.6, 10.10, 10.14 (with mergesort)
Lab 6: Advanced Functional Programming

Part VII: Object-Oriented Programming (3 lectures?)
Chapter 9
- Subtyping and Inheritance as a Type Heirachy
- Inheritance as mixed Type/Implementation Heirachy
- Single vs. Multiple Inheritance
- Method Lookup and Dynamic Binding
- Class-based vs. Object-based Languages
Lab 7: Object-Oriented AST

Part VIII: Language Support for Concurrent Programming (3 lectures?)
Chapter 12
- Implicit vs. Explicit Concurrency; Automatic Parallelization
- Shared vs. Distributed Memory Models in Languages and Computer Hardware
- Coroutines vs. True Concurrency; Generators and Filters
- Language support for Messages, Semaphores, Monitors, Initialize&Publish, Transactions
Review Questions: 12.1, 12.3, 12.18 for those who have taken CMSC356

Course review — the “same fringe” problem (1 lecture)