C S 330 - Concepts of Programng Lang

Fall 2016

Section 001: 130 MARB on M W from 1:00 pm - 2:15 pm

Instructor/TA Info

Instructor Information

Name: David Wingate
Office Phone: 801-422-6350
Email: wingated@cs.byu.edu

Name: Bryan Morse
Office Location: 3320 TMCB
Office Phone: 801-422-8146
Office Hours: Mon, Wed, Fri 11:00am-11:50am
Or By Appointment
Email: morse@byu.edu

TA Information

Name: Darin Costello
Office Location: 1152 TMCB
Office Hours: Only By Appointment
Email: dran226@gmail.com

Name: Joseph Hansen
Office Hours: Tue, Thu 1:00pm-2:50pm
Fri 12:00pm-2:55pm
Or By Appointment
Email: josephmhansen@gmail.com

Course Information

Description

The fundamental tool of a computer scientist and software engineer is the programming language. Programming languages of all shapes and sizes are found under every rock in computing. There are large, mainstream languages that everyone uses regularly. But, there are also small, domain-specific languages that are well-known to those in specific fields but not widely known to those outside.

This course will prepare you to understand the concepts underlying programming languages and to design and implement small ones. Rather than simply surveying a bunch of languages at random, we will focus on specific concepts and languages that best embody those concepts. Even if you don't use these specific languages again, you are likely to encounter the concepts in other languages, and like many things, time spent doing something a different way often causes us to be better users of our regular tools once we return to them.

We will also talk about the basics of implementing interpreters -- and language tools in general -- as well as type checking and automated memory management. Few of us will design a large mainstream language during our careers, but almost all of us will need to design and implement a small language.

Note: This course is undergoing significant revision from previous semesters, so please pardon the construction dust while we work through the changes. The schedule will certainly change as we go, and all assignments should be considered "draft" until the due date for the preceding assignment or discussed in class.

Prerequisites

This class requires CS 240 as its prerequisite (CS 142, CS 224, CS 235, CS 236), much of which we will draw from in

This class requires CS 240 and its prerequisites (CS 142, CS 224, CS 235, CS 236), each of which we will draw from in this course. You are also expected to already be a competent programmer in multiple languages--in particular, we will assume familiarity with both C++ and Java, which are taught in prerequisite courses.

Learning Outcomes

Programming Language Familiarity
Learn the vocabulary of programming language design, syntax, and semantics
Program Language Flexibility
Be able to write programs using non-imperative language paradigms.

Grading Scale

Grades	Percent
A	93%
A-	90%
B+	87%
B	83%
B-	80%
C+	77%
C	73%
C-	70%
D+	67%
D	63%
D-	60%
E	0%

Grading Policy

Each assignment must be turned in by midnight on the day it is due in order to receive full credit. Late work will be penalized 10% of the assignment maximum score per day, weekends and university holidays excepted. For example, work turned in before the deadline can receive up to 100% credit. Work turned in by midnight of the following business day can receive up to 90%, etc.
Note: The weighting of individual assignments is tentative and likely to change during the semester.

Teaching Philosophy

The best definition I ever heard of teaching is that it is creating an environment where learning can take place. Notice that the active verbs in that definition are **learning** (your job) and **creating an environment** where that can happen (our job). The learning environment for this class includes the in-class lectures, any assigned readings, homework assignments, programming labs, help sessions with the TA or instructor(s), and even (hopefully) exams. Nothing in the class is busywork. **To get the most out of the class you are encouraged to make use of all of these resources.**

Electronic Readings

Although there is no required textbook, various electronic (mainly online) readings may be assigned throughout the semester.

Software

Instructions for the software you will use during the semester (mainly compilers and interpreters for various languages) will be made as needed throughout the semester.

Coding Your Own Assignments

We strongly encourage you to get together in groups to understand concepts from class, brainstorm ideas, and work through and design algorithms. But **all code that you turn in must represent your own coding effort**. Copying code is not permitted and will be treated as cheating. If you're ever unsure where the line is between collaborating with your fellow students and writing code that reflects your own understanding and ability, please talk with one of the instructors.

If you haven't figured it out already, version control systems can be useful even when working alone. Many of you may

If you haven't figured it out already, version control systems can be useful even when working alone. Many of you may already use tools like Github or similar systems to manage your own code for your course projects. If you do so, **make sure any repositories you maintain for your code for this class are private and secure**. The university's Academic Honesty policy defines cheating as not only copying someone else's work but "allowing someone to copy from you while completing an assignment". **If your solution to an assignment for this class is found in an open online repository, it will be considered cheating and treated just the same as if you had knowingly shared your code with someone else.**

Assignments

Assignment Descriptions

Basic Racket

Aug **31** Due: Wednesday, Aug 31 at 11:59 pm

http://liftothers.org/dokuwiki/doku.php?id=cs330_f2016:racketbasics (http://liftothers.org/dokuwiki/doku.php?id=cs330_f2016:racketbasics)
Here is a basic grader to check your work:solo_grader_basic_rkt.py Download (plugins/Upload/fileDownload.php?fileId=8363a334-00di-9q7Y-oOAN-a219f9bc89e3&pubhash=F9W-uHzJ3-AxU3HJ7NpVRej3Ew16sLvqTSNfX-AWW2ojgjOGseu2K7WCw2bVCtd1Ttcf3nulbRjvNJW-C4wobQ==). This tests basic functionality and the grader we run later this week will be different. (For many future labs, there will a Docker that includes a grader and the correct version of the language.)

Lists and Recursion

Sep **07** Due: Wednesday, Sep 07 at 11:59 pm

http://liftothers.org/dokuwiki/doku.php?id=cs330_f2016:racketlists (http://liftothers.org/dokuwiki/doku.php?id=cs330_f2016:racketlists)
Here is a grader for a few generic tests: solo_grader_lists_rkt.py Download (plugins/Upload/fileDownload.php?fileId=964a569e-W7rJ-TdJc-wurX-Rua7549b8379&pubhash=0N_yy6SIUC43wtd5XjzM38WIJP1_dfvikXitgGMLCj_tNqzNhpgHYXSjZbKRePPkCbqztbKchcoiqyvVOMDgpA==).

First-class and higher-order functions

Sep **12** Due: Monday, Sep 12 at 11:59 pm

More Higher-Order Functions

Sep **14** Due: Wednesday, Sep 14 at 11:59 pm

Julia Programming 1

Sep **19** Due: Monday, Sep 19 at 11:59 pm

Interpreter 1

Sep **26** Due: Monday, Sep 26 at 11:59 pm

Interpreter 2

Oct

Oct 03 Due: Monday, Oct 03 at 11:59 pm

Interpreter 3

Oct 10 Due: Monday, Oct 10 at 11:59 pm

Interpreter 4

Oct 17 Due: Monday, Oct 17 at 11:59 pm

Logic Puzzles

Oct 24 Due: Monday, Oct 24 at 11:59 pm

Prolog 2

Oct 31 Due: Monday, Oct 31 at 11:59 pm

Erlang 1

Nov 07 Due: Monday, Nov 07 at 11:59 pm

Erlang 2

Nov 14 Due: Monday, Nov 14 at 11:59 pm

Garbage collection

Nov 21 Due: Monday, Nov 21 at 11:59 pm

Written assignment

Lazy Programming

Nov 30 Due: Wednesday, Nov 30 at 11:59 pm

Type Checker

Dec 07 Due: Wednesday, Dec 07 at 11:59 pm

University Policies

Honor Code

In keeping with the principles of the BYU Honor Code, students are expected to be honest in all of their academic work.

Academic honesty means, most fundamentally, that any work you present as your own must in fact be your own work and not that of another. Violations of this principle may result in a failing grade in the course and additional disciplinary action by the university. Students are also expected to adhere to the Dress and Grooming Standards. Adherence demonstrates respect for yourself and others and ensures an effective learning and working environment. It is the university's expectation, and every instructor's expectation in class, that each student will abide by all Honor Code standards. Please call the Honor Code Office at 422-2847 if you have questions about those standards.

Sexual Misconduct

As required by Title IX of the Education Amendments of 1972, the university prohibits sex discrimination against any participant in its education programs or activities. Title IX also prohibits sexual harassment-including sexual violence-committed by or against students, university employees, and visitors to campus. As outlined in university policy, sexual harassment, dating violence, domestic violence, sexual assault, and stalking are considered forms of "Sexual Misconduct" prohibited by the university.

University policy requires any university employee in a teaching, managerial, or supervisory role to report incidents of sexual misconduct that come to their attention through various forms including face-to-face conversation, a written class assignment or paper, class discussion, email, text, or social media post. If you encounter Sexual Misconduct, please contact the Title IX Coordinator at t9coordinator@byu.edu or 801-422-2130 or Ethics Point at https://titleix.byu.edu/report (https://titleix.byu.edu/report) or 1-888-238-1062 (24-hours). Additional information about Title IX and resources available to you can be found at http://titleix.byu.edu (http://titleix.byu.edu).

Student Disability

Brigham Young University is committed to providing a working and learning atmosphere that reasonably accommodates qualified persons with disabilities. If you have any disability which may impair your ability to complete this course successfully, please contact the University Accessibility Center (UAC), 2170 WSC or 422-2767. Reasonable academic accommodations are reviewed for all students who have qualified, documented disabilities. The UAC can also assess students for learning, attention, and emotional concerns. Services are coordinated with the student and instructor by the UAC. If you need assistance or if you feel you have been unlawfully discriminated against on the basis of disability, you may seek resolution through established grievance policy and procedures by contacting the Equal Employment Office at 422-5895, D-285 ASB.

Academic Honesty

The first injunction of the Honor Code is the call to "be honest." Students come to the university not only to improve their minds, gain knowledge, and develop skills that will assist them in their life's work, but also to build character. "President David O. McKay taught that character is the highest aim of education" (The Aims of a BYU Education, p.6). It is the purpose of the BYU Academic Honesty Policy to assist in fulfilling that aim. BYU students should seek to be totally honest in their dealings with others. They should complete their own work and be evaluated based upon that work. They should avoid academic dishonesty and misconduct in all its forms, including but not limited to plagiarism, fabrication or falsification, cheating, and other academic misconduct.

Plagiarism

Intentional plagiarism is a form of intellectual theft that violates widely recognized principles of academic integrity as well as the Honor Code. Such plagiarism may subject the student to appropriate disciplinary action administered through the university Honor Code Office, in addition to academic sanctions that may be applied by an instructor. Inadvertent plagiarism, which may not be a violation of the Honor Code, is nevertheless a form of intellectual carelessness that is unacceptable in the academic community. Plagiarism of any kind is completely contrary to the established practices of higher education where all members of the university are expected to acknowledge the original intellectual work of others that is included in their own work. In some cases, plagiarism may also involve violations of copyright law. Intentional Plagiarism-Intentional plagiarism is the deliberate act of representing the words, ideas, or data of another as one's own without providing proper attribution to the author through quotation, reference, or footnote. Inadvertent Plagiarism-Inadvertent plagiarism involves the inappropriate, but non-deliberate, use of another's words, ideas, or data without proper attribution. Inadvertent plagiarism usually results from an ignorant failure to follow established rules for documenting sources or from simply not being sufficiently careful in research and writing. Although not a violation of the Honor Code, inadvertent plagiarism is a form of academic misconduct for which an instructor can impose appropriate academic sanctions. Students who are in doubt as to whether they are providing proper attribution have the responsibility to consult with their instructor and obtain guidance. Examples of plagiarism include: Direct Plagiarism-The verbatim copying of an original source without acknowledging the source. Paraphrased Plagiarism-The paraphrasing, without acknowledgement, of ideas from another that the reader might mistake for the author's own. Plagiarism Mosaic-The borrowing of words, ideas, or

data from an original source and blending this original material with one's own without acknowledging the source. Insufficient Acknowledgement-The partial or incomplete attribution of words, ideas, or data from an original source

Insufficient Acknowledgement-The partial or incomplete attribution of words, ideas, or data from an original source. Plagiarism may occur with respect to unpublished as well as published material. Copying another student's work and submitting it as one's own individual work without proper attribution is a serious form of plagiarism.

Schedule

Date	Topic	Reading	Assignments / Exams
Week 1			
M Aug 29 Monday	**First Day of Fall Semester (08/29/2016 - 12/08/2016)** Course Introduction Racket: Introduction		
W Aug 31 Wednesday	Racket: Lists and Natural Recursion		**Basic Racket**
Week 2			
M Sep 05 Monday	**Labor Day**		
T Sep 06 Tuesday	**Add/Drop Deadline (Full Semester & 1st Term)**		
W Sep 07 Wednesday	Racket: First-class and higher-order functions		**Lists and Recursion**
Week 3			
M Sep 12 Monday	Racket: More on higher-order functions		**First-class and higher-order functions**
W Sep 14 Wednesday	Introduction to Julia		**More Higher-Order Functions**
Week 4			
M Sep 19 Monday	General structure of programming tools Interpreters: parsing, evaluation		**Julia Programming 1**
W Sep 21 Wednesday	Interpreters: expressions and conditionals		
Week 5			
M Sep 26 Monday	Interpreters: variable binding, substitution, and environments		**Interpreter 1**
W Sep 28 Wednesday	Interpreters: functions, scope, and closures		

Week 6			
M Oct 03 Monday	Interpreters: mutation and statements		**Interpreter 2**
W Oct 05 Wednesday	Program analysis		
Week 7			
M Oct 10 Monday	Program analysis (cont'd)		**Interpreter 3**
W Oct 12 Wednesday	Adding high-performance primitives		
Week 8			
M Oct 17 Monday	Catching up		**Interpreter 4**
W Oct 19 Wednesday	Prolog - declarative programming, knowledge base		
Week 9			
M Oct 24 Monday	Prolog - models of computation		**Logic Puzzles**
W Oct 26 Wednesday	Prolog - cut operator		
Week 10			
M Oct 31 Monday	Erlang - concurrency		**Prolog 2**
W Nov 02 Wednesday	Erlang - let it fail, monitoring		
Week 11			
M Nov 07 Monday	**Withdraw Deadline (Full Semester)** Continuations		**Erlang 1**
W Nov 09 Wednesday	Erlang - continuations		
Week 12			
M Nov 14 Monday	Garbage collection		**Erlang 2**
W Nov 16 Wednesday	Garbage collection (cont'd)		
Week 13			
M Nov 21 Monday	Haskell - laziness		**Garbage collection**
T Nov 22 Tuesday	**Friday Instruction**		
W Nov 23 Wednesday	**No Classes**		
Week 14			

M Nov 28 Monday	Type systems and checking		
W Nov 30 Wednesday	Type systems and checking (cont'd)		**Lazy Programming**
Week 15			
M Dec 05 Monday	Language vignettes - Wolfram Haskell - monads		
W Dec 07 Wednesday	"Growing a Language"		**Type Checker**
Th Dec 08 Thursday	**Last Day of Fall Semester (08/29/2016 - 12/08/2016)**		**All late work due**
Week 16			
T Dec 13 Tuesday	Final Exam: 3718 HBLL 2:30pm - 5:30pm		