

EECS 665: Compiler Construction

Spring 2018

Instructor:	Prof. Garrett Morris	Lab instructor:	April Wade
Email:	garrettm@ku.edu	Email:	aprilwade@ku.edu
Office:	Eaton 2028	Office:	Eaton 3025
Office hours:	Tu 12:30–14:30 Th 15:00–17:00	Office hours:	Th 13:00–14:30 Fr 12:00–13:30
Lectures:	Tu Th 11:00–12:15 2420 LEEP2	Laboratories:	Tu 9:00–10:50 Th 14:30–16:20 1005D Eaton
Course website:	http://ittc.ku.edu/~garrett/eecs665s18/		

Description. This course answers the question “What is a programming language?” from the mechanical and translational perspectives. We will discuss ways to efficiently characterize the valid terms of a programming language, both syntactically (i.e., lexing and parsing) and semantically (i.e., type checking and static analysis). We will also discuss how terms of a programming language are given meaning, via translation (i.e., compilation) to lower-level models of computation (including intermediate and machine languages). By the end of the course, you should have built a model of low-level computation, and a compilation pipeline from a high-level language targeting that model.

Recommended reading. The course is based on *Modern Compiler Implementation in ML*, by Andrew Appel. However, note that our course will differ from his in several ways. First, our programming tasks will be in Haskell, rather than in Standard ML. While there are syntactic differences between the two languages, this will hopefully not present much difficulty of understanding. There are resources on the course webpage to help you translate between ML and Haskell. Second, we will take a “ground-up” approach to building a compiler, starting at the hardware, whereas Appel starts with syntax.

Homework. This course has a significant practical component, through the laboratory assignments and 4 programming projects. You should expect these to take non-trivial amounts of time. Do not leave them until the last minute. We will attempt to grade whatever you submit; however, submitted work that does not compile can receive *at most* half of the possible credit.

The assignments and lectures will be done in the programming language Haskell. This is to make the tasks easier: tree traversals and transformations, which form the core of programming language implementation, are much more easily expressed in functional languages with pattern matching. If you are less comfortable with Haskell programming than you would prefer, there are numerous online resources you may find helpful. One online text is:

- Miran Lipovaca, *Learn You a Haskell for Great Good*, <http://learnyouahaskell.com/chapters>

The first several labs will also give you an opportunity to review Haskell programming.

Extensions. Extensions are unlikely, and will be announced in class and on the course web page should they occur. Late assignments will not be accepted.

Exams. There will be two exams, one in class on Thursday, March 15th, and one in the university-assigned finals slot from 10:30-13:00 on Wednesday, May 9th. Exams will be held in the regular class room, and are closed book and closed notes.

Grading. Grades will be based on the following breakdown, and assessed on a standard 10-point scale. *This course will not use +/- grading in Spring 2018.*

Labs	30%	A	90–100%
Homework	30%	B	80–90%
Exams	35%	C	70–80%
Midterm	15%	D	60–70%
Final	20%	F	0–60%
Quizzes	5%		

Evaluating academic work is a necessarily imprecise discipline. I will adjust the thresholds down if I think that they are not accurately reflecting student accomplishment; I will not adjust thresholds up. You must receive passing grades on *both* the homework (including the labs) and the exams (including the quizzes) to pass the course.

Course outcomes. At the end of the course, students should be capable of:

1. Understanding the role and structure of compilers, and its various phases
2. Constructing an unambiguous grammar for a programming language
3. Generating a lexer and parser using automatic tools
4. Constructing machines to recognize regular expressions (NFA, DFA) and grammars (LL and LR parsers)
5. Generating intermediate form from source code
6. Type checking and static analysis
7. Assembly/binary code generation

Honesty and academic misconduct. The work you submit in this course should be yours, and yours alone. You are encouraged to discuss course materials, including homework problems, with the other students in the class. However, you should only submit work that is entirely yours, and has not been derived from other sources or been shared with other students. Submitting work that is not yours is academic misconduct, and will result in receiving a score of 0 on the assignment or exam, and being reported to the chair of the department. You should be aware of the university's academic misconduct policies:

<http://policy.ku.edu/governance/USRR#art2sect6>.

There is an ever escalating arms race between the makers of automated systems for detecting cheating and the efforts of some students to get around them. I suggest that at this point it is probably easier to get passing grades honestly than dishonestly.

Accommodation procedure. The Academic Achievement and Access Center (AAAC) coordinates academic accommodations and services for all eligible KU students with disabilities. If you have a disability for which you wish to request accommodations and have not contacted the AAAC, please do so as soon as possible. They are located in 22 Strong Hall and can be reached at 785-864-4064 (V/TTY). Information about their services can be found at <http://www.access.ku.edu>. Please contact me privately in regard to your needs in this course.

Nondiscrimination. The University of Kansas prohibits discrimination on the basis of race, color, ethnicity, religion, sex, national origin, age, ancestry, disability, status as a veteran, sexual orientation, marital status, parental status, retaliation, gender identity, gender expression and genetic information in the University's programs and activities. Please contact the University's Title IX Coordinator at IOA@ku.edu with any inquiries.

Religious observances. Should the examination schedule for this course conflict with your mandated religious observance, please contact me at the *beginning* of the semester so that we can schedule a make-up exam at a mutually acceptable time. In addition, students will not be penalized for absence from regularly scheduled class activities which conflict with mandated religious observances. Students are responsible for initiating discussion with the instructor to reach a mutually acceptable solution.

Concealed carry. Individuals who choose to carry concealed handguns are solely responsible to do so in a safe and secure manner in strict conformity with [state and federal laws](#) and [KU weapons policy](#). Safety measures outlined in the KU weapons policy specify that a concealed handgun:

- Must be under the constant control of the carrier.
- Must be out of view, concealed either on the body of the carrier, or backpack, purse, or bag that remains under the constant control of the carrier.
- Must be in a holster that covers the trigger area and secures any external hammer in an un-cocked position
- Must have the safety on, and have no round in the chamber.

Instructors are allowed by Kansas Board of Regents policy, to require backpacks, purses and other bags be placed at the front of the room during exams, and as such those items will not be under the constant control of the individual. Students who choose to carry a concealed handgun in a purse, backpack, or bag must review and plan each day accordingly, and are responsible for making alternate arrangements as necessary. The university does not provide appropriate secured storage for concealed handguns.

Individuals who violate the KU weapons policy may be asked to leave campus with the weapon and may face disciplinary action under the appropriate university code of conduct.