2017 Spring - 406 - Compiler Construction - Syllabus

This class is taught by Jay McCarthy. Call him Jay. Email him at `first-name DOT last-name AT gmail DOT com`.

We meet in Olsen 401 at 1100-1215.

Jay's office hours are TR 0730-1230 in Olsen 221.

There is a mailing list hosted at Google Groups. Use it to ask non-revealing questions and receive answers, as well as general course announcements. You are responsible for reading the content of this mailing list.

1 Important Course Details

Includes both theory and practice. A study of grammars, specification and classes, the translation pipeline: lexical analysis, parsing, semantic analysis, code generation and optimization, and syntax-directed translation. Use of automatic generation tools in the actual production of a complete compiler for some language.

This course provides the following Essential Learning Outcomes (ELOs):

Information Literacy (IL): The ability to use digital technologies, communication tools and/or networks to define a problem or an information need; devise an effective search strategy;identify, locate, and evaluate appropriate sources; and manage, synthesize, use and effectively communicate information ethically and legally.

Applied and Integrative Learning (AIL): Applied and Integrative Learning is an understanding and disposition that a student builds across curriculum and co-curriculum, fostering learning between courses or by connecting courses to experiential learning opportunities.

Readings. (EC) We'll be using the book Essentials of Compilation, by Jeremy Siek. For your convenience, I have included a copy of the book in PDF form on this site. I highly recommend this guide to X86-64 assembly.

Policy. We do not allow electronic devices, such as laptops, phones, and tablets, to be used during class, without an explicit accomodation exception. I will first ask you to put it away and then I will ask you to leave.

2 Lectures, Assignments, and Exercises

Day	Date	Topic	Notes
1	Tue 01/17	Basics	EC 2 **Links:** 1.pdf
2	Thu 01/19	Basics	EC 2 **Links:** 2.pdf
3	Tue 01/24	Basics	EC 2 **Exercise:** basics **Links:** 3.pdf
4	Thu 01/26	Register Allocation	EC 3

				Links: 4.pdf
5	Tue	01/31	Register Allocation	EC 3 **Links:** 5.pdf
6	Thu	02/02	Register Allocation	EC 3 **Exercise:** regalloc
7	Tue	02/07	Control Flow	EC 4 **Links:** 7.pdf
8	Thu	02/09	Control Flow	EC 4
9	Tue	02/14	Control Flow	EC 4 **Exercise:** control **Links:** 9.pdf
10	Thu	02/16	Heap Allocation and GC	EC 5 **Links:** 10.pdf
	Tue	02/21	No class	
11	Thu	02/23	Heap Allocation and GC	EC 5 **Links:** 11.pdf
12	Tue	02/28	Heap Allocation and GC	EC 5 **Exercise:** heap **Links:** 12.pdf
13	Thu	03/02	Midterm Review	**Due:** Paper
14	Tue	03/07	Midterm	
15	Thu	03/09	Functions and Closures	EC 6 **Links:** 15.pdf
	Tue	03/14	No class	
	Thu	03/16	No class	
16	Tue	03/21	Functions and Closures	EC 6 & 7 **Links:** 16.pdf
17	Thu	03/23	Functions and Closures	EC 7 **Exercise:** clo
18	Tue	03/28	High-Level Optimization	EC 11 (see Waddel 1997) **Links:** 18.pdf
19	Thu	03/30	High-Level Optimization	EC 11 **Links:** 19.pdf
20	*Tue*	*04/04*	High-Level Optimization	*Jay will be away.* EC 11 **Exercise:** opt **Links:** 20.pdf
21	Thu	04/06	Macro Expansion	Naive Expansion by Example **Links:** 21.pdf
22	Tue	04/11	Macro Expansion	Hygeine through Marks **Links:** 22.pdf
23	Thu	04/13	Macro Expansion	Sets of Scopes **Exercise:** macro **Links:** 23-Talk 1.mov 23-Talk 2.mov
24	Tue	04/18	Compilation Tools	LLVM **Links:** LLVM Manual
25	Thu	04/20	Compilation Tools	Parser Generators

				Links: 25-expanded.rkt 25.rkt
26	Tue	04/25	Compilation Tools	Managed Runtimes **Exercise:** tools **Links:** 26.pdf
27	Thu	04/27	Final Review	**Due:** Presentation
Final	Thu	05/04	Final	**We will meet at 1130-1430.**

3 Work in this Course

If you do a little work for this course every day, including reading the textbook and doing the exercises, you will find the exams and papers easy to write. If you just search for answers in course material as you try to complete work, you will not do well.

I highly recommend that you read this article about grading. I also recommend you read this article about the stress that you may experience in a computer science program. Please try to make healthy productive choices in your life. I would love the opportunity to help you in any ways I can.

3.1 Exercises

In the Lectures, Assignments, and Exercises table above, a number of exercises are referenced. You are not required to do them, but I highly recommend that you complete these. Exams may reference material from class, from the reading, or from these exercises. If you do not do the exercises, then there will be exam questions which will be unintelligible and nigh impossible for you to answer successfully. If you would like to discuss your solution to them, please meet me in my office. I will not review solutions via email, but I will answer clarifying questions.

3.2 Paper

I will not review or discuss your paper via email, but I am happy to discuss it with you in my office.

You hand in your paper by putting them on the lectern at the start of class on the day they are due. No papers will be accepted at any other time, for any reason. No papers will be acceped electronically. If you cannot turn in your work personally, you may have a friend do it.

Your turn-in must clearly have your name on it and indicate which option you chose.

Your turn-in must be prepared electronically. I recommend using LaTeX. You must use 0.5" margins, the font Times New Roman at 11pt, and single-spacing. Each page must be numbered in the footer. You must print with black ink (color may be used in diagrams, but not in the text) on white paper. If your printer is low on ink or toner, find another one rather than giving me something that is faded. I recommend using only mild formating. You should not include long quotes, code samples, or large diagrams. You should reference sources using the BibTeX `alpha` style.

Your turn-in must be stapled together. It is your responsible to ensure this is the case.

You may not turn in more than five pieces of paper. You probably want to turn in exactly five pieces of paper. This means ten "pages" of content, although since references count it may be more like nine.

Work that is incorrectly formatted will be returned ungraded and I will consider it as though you never turned it in.

Your paper will receive a grade from the closed interval between `0` and `1`. I will evaluate your paper based on the veracity of its content, its logical structure, as well as its grammar, style, and form.

3.3 Presentation

I will not review or discuss your presentation via email, but I am happy to discuss it with you in my office, although naturally it is hard to separate the actual presentation from the planning of it, so you are mostly on your own.

You must schedule an appointment with me to meet in my office to give your presentation. I will not remind you of this requirement or of when your appointment is. You will only have one chance to give your presentation and only at the time you originally scheduled. You should plan to present for about fifteen minutes.

You may use a whiteboard and markers or a notepad and pen for your presentation.

You may not use any computer software during the presentation, although if you are presenting code, etc, you can email it to me beforehand and I can have it available for you.

You may bring one 3" by 5" ruled index card of notes, if you would like.

Your presentation will receive a grade from the closed interval between `0` and `1`. I will evaluate your presentation based on the veracity of its content, its logical structure, as well as its style and form.

3.4 Exams

Exams must be taken in person, at the scheduled time.

Exams must be completed without any notes, books, or resources of any kind.

You must use 8.5" by 11" College Ruled paper without frill and write using a black pen. You are responsible for my ability to read and understand your writing and diagrams.

Your exams must clearly have your name on it.

You may not hand in more than five pages and individual answers must not use more than a single piece of paper (front and back.) These pages must be stapled together, which you are responsible for. I recommend pre-stapling your paper before the exam.

Exams that are incorrectly formatted will be returned ungraded and I will consider it as though you never turned it in.

Your exams will receive a grade from the closed interval between `0` and `1.25`. I will evaluate your exam based on the veracity of its content and the logical structure of

your answers.

In your exam, you will be provided four groups of two questions. You can answer one question from each group. After you answer one question from each group, you may answer a single other question (i.e. answering two questions for one group.) If you provide answers for more than five questions or two for more than one group, then I will return your exam ungraded as though you never turned it in. Each question will be equally weighted at `0.25`.

Nearly all exam questions are best answered using all of the available space. If you can't write more than a paragraph, then you probably don't really understand the question well enough to answer it correctly.

3.5 Class Numeric Grade

This function takes your turn-ins and combines them into a numeric grade:

```
> (define (combine paper midterm pres final)
    (+ (* 0.2 paper) (* 0.2 midterm)
       (* 0.3  pres) (* 0.3   final)))
```

Examples:

```
> (combine 0.0 0.0 0.0 0.0)
0.0
> (combine 0.5 0.5 0.5 0.5)
0.5
> (combine 0.0 0.0 1.0 1.0)
0.6
> (combine 0.0 0.0 1.0 1.25)
0.675
> (combine 0.0 1.25 0.0 1.25)
0.625
> (combine 1.0 1.0 1.0 1.0)
1.0
> (combine 1.0 1.25 1.0 1.25)
1.125
```

This function is designed to allow you to perform poorly at the beginning of the course, but still recover and end with a good grade.

3.6 Course Letter Grade

The following function to converts your numeric grade into a letter grade:

```
> (define (convert-to-letter ng)
    (cond
      [(> ng 0.93) "A"]
      [(> ng 0.9) "A-"]
      [(> ng 0.86) "B+"]
      [(> ng 0.83) "B"]
      [(> ng 0.8) "B-"]
      [(> ng 0.76) "C+"]
      [(> ng 0.73) "C"]
      [(> ng 0.7) "C-"]
      [(> ng 0.66) "D+"]
      [(> ng 0.6) "D"]
      [else "F"]))
```

Examples:

```
> (convert-to-letter 1)
"A"
> (convert-to-letter 0.94)
"A"
> (convert-to-letter 0.899999)
"B+"
> (convert-to-letter 0.81)
"B-"
> (convert-to-letter 0.74)
"C"
> (convert-to-letter 0.6999999)
"D+"
> (convert-to-letter 0.62)
"D"
> (convert-to-letter 0.57)
"F"
```

3.7 Graduate Student Option

If you are taking the class as a graduate class, then you must do either two paper options or two presentation options. The extra assignment you do will be due by the date of the final. In addition, your work will be graded to a higher standard than the other students.

4 Help

My job is to help you.

One of the best ways for me to help you is to teach you how to help yourself. The main way I will put this idea into practice is that I will not answer your question until you have shown that you've read the relevant part of the textbook and have a concrete question or point of confusion about it.

If you need a "shallow" amount of help, then look at the Google Group. First, see if I have already answered your question. Then, send your own email.

Only send me personal email if you need to talk about something private, such as your grades. Anything else is best discussed in public, so others can benefit. If you do send personal email, put `[CS406]` as a prefix in the subject.

If you need a "deep" amount of help, please come to my office or call me (801-361-0732) and we'll talk and try to resolve whatever ails you.

5 Fine Print

In this course, all work is to be each student's own. Students should therefore be familiar with the University's rules on academic dishonesty, which can be found in the Bulletin of Undergraduate Studies and in the Schedule of Classes. In particular, plagiarism will not be tolerated! Any student caught plagiarizing another's work will automatically receive a grade of F for the course. If you are unsure as to what constitutes plagiarism, it is your responsibility to check with the instructor. Other forms of dishonesty will result in similar actions. You may collaborate with your classmates on the design and results of the programs you will write in this course, but each student must implement these programs alone. Submission of shared student code is not permissible, and will result in a grade of F for the course. Help files are typically provided for each programming assignment, and students are encouraged to cut

and paste useful code from these help files into their assignment submissions, but all other code must be the specific work of each student.

You are not allowed to post solution code to problem sets assigned in this class in public places (e.g. Github). This includes your own solutions as well as solutions that may be provided by the instructors.

This policy is a courtesy to future students, who — to the fullest extent possible — should have the opportunity to struggle with the problems in the same way that you do.

Non-compliance will be pursued rigorously per UMass Lowell's academic integrity policy.