THE UNIVERSITY OF ARIZONA

SYLLABUS

CSc 553, Principles of Compilation

Fall 2018

MW 2:00-3:15, Bio Sciences West 237

Class website: http://www.cs.arizona.edu/classes/cs553/fall18/

Description of Course

This course considers advanced topics in compilation. Specific topics discussed will include: intermediate program representations; code generation and machine-independent peephole optimization; register allocation; formation and solution of dataflow analysis problems; code optimization using data flow information; memory hierarchy optimizations and profile-guided code placement; instruction scheduling.

.

Course Prerequisites

Courses: CSc 453 and CSc 473 (or their equivalents).
Knowledge: Proficiency in programming in C. Familiarity with software development in a Unix/Linux environment, including tools such as editors, debuggers, **make**, etc.

Contact Information

Instructor:

Saumya Debray
Office: Gould-Simpson 735
Phone: (520) 621-4527; Email: debray@cs.arizona.edu

Course Format

The course will consist of two weekly in-class lectures (MW 2:00-3:15).

Course Objectives and Expected Learning Outcomes

Students who successfully complete this course should be able to understand:

- intermediate representations (IRs) used by compilers, including: abstract syntax trees; three-address code; basic blocks and control flow graphs; and static single-assignment form (SSA);
- code generation via syntax-directed translation;
- program analysis: control-flow analysis and data-flow analysis; formulation and iterative solution of data-flow equations;
- machine-independent and machine-dependent code optimization; memory hierarchy optimization; profile-guide code placement; instruction scheduling;
- (as time permits) just-in-time compilation and dynamic code optimization.

As part of the course, students will implement an optimizing compiler for a significant subset of C and also study research papers from current research literature.

Absence and Class Participation Policy

Attendance will be expected, but not recorded. Students are fully responsible for all material presented or assigned in class. For this reason, and because participating in the course and attending lectures and other course events are vital to the learning process, attendance is strongly recommended.

Students who miss exams or assignment submission deadlines due to illness or emergency are required to bring documentation from their health-care provider or other relevant professional third parties.

The UA's policy concerning Class Attendance, Participation, and Administrative Drops is available at: http://catalog.arizona.edu/policy/class-attendance-participation-and-administrative-drop

The UA policy regarding absences for any sincerely held religious belief, observance or practice will be accommodated where reasonable, http://policy.arizona.edu/human-resources/religious-accommodation-policy.

Absences pre-approved by the UA Dean of Students (or Dean Designee) will be honored. See: https://deanofstudents.arizona.edu/absences

Makeup Policy for Students Who Register Late

Students who register after the first class meeting may not make up any missed work.

Course Communications

Course communications will be made through:

- D2L (for announcements)
- Piazza (for questions and discussions). The Piazza website for this class is https://piazza.com/arizona/fall2018/csc553/home. Students are responsible for enrolling for this class in Piazza.

Required Texts or Readings

This class has no required text. Technical papers from the literature will be assigned during the course of the semester, and the instructor's lecture notes (on the class website noted above) will be available as a reference.

Required or Special Materials

None

Required Extracurricular Activities (if any)

None

Assignments and Examinations

Programming Assignments

The course has a programming project where students implement an optimizing compiler for a subset of the C programming language. This project is divided into a number of programming assignments, each implementing a distinct component of an optimizing compiler.

Collaboration: All assignments are individual (i.e., there is no partnering).

Timeliness: Assignments are due at the time stated on the assignment spec. Late submissions are permitted for up to 5 late days per assignment, but will be assessed a late penalty of 10 points per day.

You may turn in multiple submissions for the same assignment. Each day's submission will be graded separately, and you will be awarded the maximum of their scores (after adjusting for late penalties where appropriate). *Note that for each day's submissions, later submissions will overwrite files submitted earlier.*

Schedule: The schedule for these assignments is as follows (Note: this is subject to change with advance notice):

Assg No.	Topic	Start date	Due date
1	Syntax tree traversal	8/27/2018	9/3/2018
2	Code generation	9/10/2018	10/1/2018
3	Machine-independent optimization	10/8/2018	10/22/2018
4	Register allocation	10/29/2018	11/12/2018
5	Optimization/term paper	11/19/2018	12/3/2018

Exams

Exams will be held in the regular classroom for the class (i.e., BioSciences West 237) unless explicitly noted. The schedule for exams is as follows:

- Midterm: Mon Oct 15, 2018: 2:00 – 3:15pm
- Final: Wed Dec 12, 2018: 1:00 – 3:00pm

Without prior arrangements, missed exams result in a grade of zero. If you will be absent on the date of an exam due to religious reasons or because of a pre-approved absence by the Dean of Students, please contact me ahead of time to work out an alternative time for your exam.

Final Examination

Wed Dec 12, 2018: 1:00 – 3:00pm.

Final Exam Regulations: see https://www.registrar.arizona.edu/courses/final-examination-regulations-and-information
Final Exam Schedule: see http://www.registrar.arizona.edu/schedules/finals.htm

Grading Scale and Policies

Grades will be computed based on the following weights for the various components of the class:

Midterm exam	20%
Final exam	30%
Programming project and/or term paper	50%

Your grade will be determined by the overall weighted average of your scores, computed using the weights given above, based on the following mapping:

Weighted average	Grade
≥ 90	A
≥ 80 but below 90	B
≥ 65 but below 80	C
≥ 55 but below 65	D
< 55	E

University policy regarding grades and grading systems is available at http://catalog.arizona.edu/policy/grades-and-grading-system

Grading procedure

The size of the programs involved in the project makes it impractical to manually examine your source code to determine its correctness. Instead, we will use the following procedure:

- You will develop and test your code using your own test cases. It is permissible for students to share test cases.
- I will make my test cases public after the submission deadline.
- Your code will be graded on my test cases using a grading script. You will be awarded a preliminary score based on the number of test cases failed.
- If you were penalized more than once for the same problem, you will have the option of bringing this to my attention over the two weeks following notification of your preliminary score. Specifically, you will need to provide me with the following items (you can do this either via email or in person):
    - a list of the specific problems in your code; and
    - for each problem, the test cases that failed as a result.
- Based on this, I may adjust your preliminary score where appropriate, based on my assessment of the seriousness of the problems. *However, any such adjustment will always be positive, i.e., you will not be penalized—but may be rewarded—for identifying and explaining the problems in your code.*

Requests for incomplete (I) or withdrawal (W) must be made in accordance with University policies, which are available at http://catalog.arizona.edu/policy/grades-and-grading-system#incomplete and http://catalog.arizona.edu/policy/grades-and-grading-system#Withdrawal respectively.

Dispute of Grade Policy: All regrade requests for programming assignments must be made within two weeks of when the grade is returned. All regrade requests for midterm exams must be made within one week of when the graded exam is returned.

Scheduled Topics/Activities

Week no.	Week of	Lecture Topic	Assignments and Exams
1	8/20/2018	Overview	
2	8/27/2018	Syntax trees and symbol tables; Intermediate code generation	Assg 1 start: 8/27
3	9/3/2018	Code generation	Assg 1 due: 9/3
4	9/10/2018	Code generation	Assg 2 start: 9/10
5	9/17/2018	Control flow analysis	
6	9/24/2018	Dataflow analysis, code optimization	
7	10/1/2018	Dataflow analysis, code optimization	Assg 2 due: 10/1
8	10/8/2018	Dataflow analysis, code optimization	Assg 3 start: 10/8
9	10/15/2018	Register allocation	**Midterm: 10/15**
10	10/22/2018	Register allocation	Assg 3 due: 10/22
11	10/29/2018	Instruction scheduling	Assg 4 start: 10/29
12	11/5/2018	SSA representation	
13	11/12/2018	profile-guided code optimization	Assg 4 due: 11/12
14	11/19/2018	Dynamic analysis: profiling	Assg 5 start: 11/19
15	11/26/2018	Dynamic analysis: JIT compilation	
16	12/3/2018	Dynamic analysis: JIT compilation	Assg 5 due: 12/3
			Final exam: 12/12

Department of Computer Science Code of Conduct

The Department of Computer Science is committed to providing and maintaining a supportive educational environment for all. We strive to be welcoming and inclusive, respect privacy and confidentiality, behave respectfully and courteously, and practice intellectual honesty. Disruptive behaviors (such as physical or emotional harassment, dismissive attitudes, and abuse of department resources) will not be tolerated. The complete Code of Conduct is available on our department web site. We expect that you will adhere to this code, as well as the UA Student Code of Conduct, while you are a member of this class.

Classroom Behavior Policy

To foster a positive learning environment, students and instructors have a shared responsibility. We want a safe, welcoming, and inclusive environment where all of us feel comfortable with each other and where we can challenge ourselves to succeed. To that end, our focus is on the tasks at hand and not on extraneous activities (e.g., texting, chatting, reading a newspaper, making phone calls, web surfing, etc.). Activities that are disruptive or distracting to others, such as those listed below, will not be tolerated in the classroom:

- Reading newspapers or magazines.
- Discussions of extended duration (except when you've been asked to discuss something as part of a class activity).
- Phone conversations, texting.
- Eating meals.

Students observed engaging in disruptive activity will be asked to cease this behavior. Those who continue to disrupt the class will be asked to leave lecture or discussion and may be reported to the Dean of Students.

Threatening Behavior Policy

The UA Threatening Behavior by Students Policy prohibits threats of physical harm to any member of the University community, including to oneself. See http://policy.arizona.edu/education-and-student-affairs/threatening-behavior-students.

Accessibility and Accommodations

At the University of Arizona we strive to make learning experiences as accessible as possible. If you anticipate or experience physical or academic barriers based on disability or pregnancy, you are welcome to let me know so that we can discuss options. You are also encouraged to contact Disability Resources (520-621-3268) to explore reasonable accommodation.

If our class meets at a campus location: Please be aware that the accessible table and chairs in this room should remain available for students who find that standard classroom seating is not usable.

Code of Academic Integrity

Students are encouraged to share intellectual views and discuss freely the principles and applications of course materials. However, graded work/exercises must be the product of independent effort unless otherwise instructed. Students are expected to adhere to the UA Code of Academic Integrity as described in the UA General Catalog. See: http://deanofstudents.arizona.edu/academic-integrity/students/academic-integrity.

Selling class notes and/or other course materials to other students or to a third party for resale is not permitted without the instructor’s express written consent. Violations to this and other course rules are subject to the Code of Academic Integrity and may result in course sanctions. Additionally, students who use D2L or UA e-mail to sell or buy these copyrighted materials are subject to Code of Conduct Violations for misuse of student e-mail addresses. This conduct may also constitute copyright infringement.

Programming assignments: what’s allowed and what isn’t

It is permissible to discuss problems with others in broad terms, e.g., the structure or approach of a program. It is not permissible to discuss concrete details of solutions to a particular assignment before the due date/time for that assignment. In other words, you can talk to each other in English, but not in C/Unix.

The work you turn in for credit should be substantially your own. It is permissible to share test inputs with other students; collaboration beyond this on programming assignments is not permitted.

It is permissible to use modest amounts of "publicly visible" code — code that is available in books or magazines, or which has been distributed/discussed in class — in programming assignments, *as long as the authorship of such code is adequately and explicitly acknowledged*. It is not permissible to solicit code from others. It is also not permissible to use code written by students in previous terms. Please check with me ahead of time if you'd like to use someone else's code in order to make sure that the amount of code is indeed modest.

For the purposes of this course, cheating is considered to be any attempt to pass off someone else's work as your own. Cheating will not be tolerated: any student caught cheating or helping another student cheat in homeworks, exams, or programming assignments, will be given a failing grade in the course. I intend to interpret the phrase "helping another student cheat" broadly: for example, if another student gains access to your code because you forgot to logout, or were careless about listings that were dumped into the recycling bin, you have helped that student cheat. For the same reason, you should be very careful about posting your code to publicly visible media, e.g., Piazza or Github.

UA Nondiscrimination and Anti-harassment Policy

The University is committed to creating and maintaining an environment free of discrimination; see http://policy.arizona.edu/human-resources/nondiscrimination-and-anti-harassment-policy

Our classroom is a place where everyone is encouraged to express well-formed opinions and their reasons for those opinions. We also want to create a tolerant and open environment where such opinions can be expressed without resorting to bullying or discrimination of others.

Additional Resources for Students

UA Academic policies and procedures are available at http://catalog.arizona.edu/policies

Student Assistance and Advocacy information is available at http://deanofstudents.arizona.edu/student-assistance/students/student-assistance

Confidentiality of Student Records

http://www.registrar.arizona.edu/personal-information/family-educational-rights-and-privacy-act-1974-ferpa?topic=ferpa

Subject to Change Statement

Information contained in the course syllabus, other than the grade and absence policy, may be subject to change with advance notice, as deemed appropriate by the instructor.