

CS 5363: PROGRAMMING LANGUAGES AND COMPILERS
FALL 2009, COURSE SYLLABUS

CS 5363: Tues./Thur. 7:00-8:15pm in SB 2.02.02

Instructor

Instructor: Dr. Jeffery von Ronne
Phone: 210-458-5667
Email: vonronne@cs.utsa.edu
Office Location: S.B. 4.01.14
Office Hours: Wed. 11:00-12:00pm, Thur. 11:00-12:00pm, and by appointment

TA: Hongwei Tian
TA Email: htian@cs.utsa.edu

Course Description and Goals

A study of programming languages with an emphasis on their implementation. Topics include lexical analysis, language syntax, control structures, the binding of names, procedures, and their implementation in compilers.

The goals of this course are for students. . .

1. to have an opportunity to experience a substantial programming project,
2. to understand how programs are implemented using compilers,
3. to develop a conceptual framework in which to students can compare and evaluate programming language features and paradigms, and
4. to become familiar with a substantial portion of the material that may appear on the PhD qualifying exam for Programming Languages and Compilers.

Prerequisites

The prerequisites for this course are:

1. CS 2233: Discrete Mathematical Structures
2. CS 3343: Analysis of Algorithms

The course work will involve writing a complete compiler. This will require the ability to take a description of an algorithm or data structure, and then implement it and integrate with the other compiler components. Students with limited programming experience are encouraged to consider whether they might be better off in a different course.

Textbooks

There are two textbooks for this course. The first is Michael Scott's "Programming Language Pragmatics, 3rd Ed." (2009), and the second is Keith Cooper and Linda Torzon's "Engineering a Compiler" (2004). The second edition of "Programming Language Pragmatics" should also be acceptable. In addition, research papers on compilers and programming languages may be provided to supplement and the textbook's coverage of certain topics.

Course Outline

The topics to be covered this semester are expected to include:

1. Overview of Compilation and Interpretation (Scott, Ch. 1; Cooper, 1.4/1.5)
2. Programming Language Syntax (Cooper, Ch. 2, 3)
3. Attribute Grammars (Scott Ch. 4)
4. Intermediate Representations (Cooper, Ch. 5)
5. Activation Records and Code Shape (Cooper, Ch. 6, 7)
6. Machine-Independent Optimizations (Cooper, Ch. 8, 10)
7. Machine-Dependent Optimizations (Cooper, Ch. 11, 12, and 13)
8. Data-Flow Analysis (Cooper, Ch. 9)
9. Data Types (Scott, Ch. 7)
10. Programming Language Paradigms (Scott, Ch. 9, 10)

Assignments, Exams, and Grading

The course work will consist of several homework assignments, a multi-phase compiler project, a midterm exam, and a final exam.

Students whose weighted average exceeds 60% across the following components **will receive at least a 'D':**

- 10% compiler project (phase I)
- 50% final exam
- 25% midterm exam
- 15% homework

Students whose weighted average exceeds 70% across the following components **will receive at least a 'C':**

- 10% compiler project (phase I)
- 50% final exam
- 25% midterm exam
- 15% homework

Students who meet the requirements for a 'C' and whose weighted average exceeds 72.5% across the following components **will receive at least a 'B':**

- 40% compiler project (phase I + core)
- 33.3% final exam
- 16.7% midterm exam
- 10% homeworks

Students who meet the requirements for a 'B' and whose weighted average exceeds 75% across the following components **will receive an 'A':**

- 55% compiler project (phase I + core + extensions)
- 25% final exam
- 12.5% midterm exam
- 7.5% homeworks

The instructor may, after reviewing class performance and specific grading criteria for individual components, adopt more lenient criteria for each grade.

Course web page and Newsgroup

Information related to the course (including project information, homework assignments, tentative schedule, and exam dates) will be posted periodically at <http://www.cs.utsa.edu/~vonronne/classes/cs5363-f09>. The newsgroup `utsa.cs.5363.d` is available for discussion.

Policies

As a matter of policy, **late assignments will not be accepted**, but partial credit will be given for partially complete/working assignments. Exceptions may be made on a case by case basis.

Students are **expected to attend all lectures**. Although absences will not result in a direct grade penalty, students are responsible for knowing all material provided and announcements made during lectures. If a student finds it necessary to miss a lecture, they should find out what was covered from a classmate.

Scholastic Dishonesty

Unless stated otherwise in the instructions for a particular assignment, students may collaborate with each other on the homework assignments, but **must complete compiler projects and exams independently**.

For the project, it is OK to discuss the assignment, strategies, algorithms, and compiler error messages with your classmates in general terms, but you should not be discussing the details of, looking at or copying each others code. Your code should be your own and not look like anyone else's. It is, however, permitted to re-implement known/published algorithms (from textbooks or research papers) in the context of your own compiler, but you should cite the source for any such algorithms not found in your text book. In any case, you must understand everything, and you are expected to be able to explain the purpose of every line of code that you turn in.

If students collaborate on homework assignments, they should only submit solutions they understand and list everyone they collaborated with.

Cases of academic dishonesty will be referred to Student Judicial Affairs. The UTSA's Student Code of Conduct (<http://www.utsa.edu/infoguide/appendices/b.html>) contains a description of expected behavior and the potential penalties for scholastic dishonesty.

Disability Services

Students who have a disability that may impact your classroom performance are encouraged to contact Disability Services (<http://www.utsa.edu/disability/>). If accommodations are determined to be appropriate, please notify the instructor as early as possible.

Drop Date

September 1 is the last day to add or drop courses through ASAP. September 11 is the census date; after this date additional university approval/justification is needed to add a course; it is also the last date to withdraw from a class and receive a refund and no grade. October 16 is the last to drop this course with a grade of W.