

# Syllabus

---

**Instructor.** Aggelos Kiayias.

Office: ITE Building 243.

Office Hours: Tuesday 3:30pm-5pm, Wednesday 5-7pm and by appointment.

Contact Information: <http://www.cse.uconn.edu/~akiayias/>

**TA.** David Walluck. Send e-mail to [walluck@cse.uconn.edu](mailto:walluck@cse.uconn.edu). Lab: ITE Bldg 215

**Textbook:** Compilers: Principles, Techniques, and Tools, by A. V. Aho, R. Sethi, J. D. Ullman

**Class Web-site.** (for class notes, homeworks/project descriptions and general info)

<http://www.cse.uconn.edu/~akiayias/cse244fa06/>

**Class Bulletin Board.** (for announcements, communication with the course staff and discussions)

<http://cryptodrm.engr.uconn.edu/cse244fa07bb/>

**Assignments.** There will be six assignments all culminating to a large project (50% of the grade). Five homeworks will be mandatory and one is extra credit (homework 6).

**Handing your work.** The submission of assignments will be done entirely by e-mail. The e-mail address for this purpose is

[c244fa07@cse.uconn.edu](mailto:c244fa07@cse.uconn.edu)

Each homework is due on **midnight** on the due date.

**Lateness.** Throughout the semester you have 10 late days that you can use to delay the due date of your homework. Due dates are calculated based on the time that your assignment arrives on the above address.

**Tests.** There will be 2 tests (midterm and final) counting for 50% of the grade. The exams will be open book.

**Collaboration.** The idea of the class is that you work on your own compiler using some pieces of source code provided and the skills you acquire from the lectures. You may discuss the assignments with your fellow students and you are encouraged to use the bulletin board of the class for this purpose. On the other hand, sharing your code or debugging your code together with other students is prohibited. Feel free to ask the instructor if you are not sure regarding collaboration policies.

**What is this class.** This class is a hands-on introduction to compiler design.

**What will be covered.** The subjects that we will cover throughout the semester include: lexical analysis, top down and bottom up parsing, abstract syntax trtees, attributed grammars, dealing with grammar ambiguities, semantic analysis, intermediate code generation, runtime organization, control flow graphs, machine code generation, register allocation, compiler optimization and security in compiler design.

**Tentative schedule.** The schedule is tentative and subject to changes over the course of the semester. For more information please consult the web-page of the class and the bulletin board.

| Lecture | Date       | Topic                      | Reading          | Homeworks      |
|---------|------------|----------------------------|------------------|----------------|
| 0       | 08/28/2007 | Introduction               | Chapter 1,2      |                |
| 1       | 08/30/2007 | Lexical Analysis I         | Chapter 3        |                |
| 2       | 09/04/2007 | Lexical Analysis II        | Chapter 3        |                |
| 3       | 09/06/2007 | Introduction to Parsing I  | Chapter 4 S1-S3  | Out(1)         |
| 4       | 09/11/2007 | Introduction to Parsing II | Chapter 4 S5-7   |                |
| 5       | 09/13/2007 | Top-Down Parsing I         | Chapter 4 S5-7   | Due(1)         |
| 6       | 09/18/2007 | Top-Down Parsing II        | Chapter 4 S4     |                |
| 7       | 09/20/2007 | Bottom Up Parsing Intro    | Chapter 4 S4     |                |
| 8       | 09/25/2007 | Bottom Up Parsing I        | Chapter 4 S4     | Out(2)         |
| 9       | 09/27/2007 | Bottom Up Parsing II       | Chapter 4 S8 , 5 |                |
| 10      | 10/02/2007 | Attributed Grammars        | Chapter 5        | Due(2), Out(3) |
| 11      | 10/04/2007 | Semantic Analysis I        | Chapter 6        |                |
| 12      | 10/09/2007 | Semantic Analysis II       | Chapter 6        | Due(3)         |
| 13      | 10/11/2007 | <i>Midterm</i>             |                  |                |
| 14      | 10/16/2007 | Runtime Organization I     | Chapter 7        | Out(4)         |
| 15      | 10/18/2007 | Runtime Organization II    | Chapter 7        |                |
| 16      | 10/23/2007 | Intermediate Code          | Chapter 8        |                |
| 17      | 10/25/2007 | IR Generation              | Chapter 8        |                |
| 18      | 10/30/2007 | Control Flow Graphs        | Chapter 9 S1-S4  | Due(4)         |
| 19      | 11/01/2007 | Machine Code               | Chapter 9        | Out(5)         |
| 20      | 11/06/2007 | MC Generation              | Chapter 9        |                |
| 21      | 11/08/2007 | Register Allocation        | Chapter 9 S7-S8  |                |
| 22      | 11/13/2007 | Dynamic Code               | Chapter 9 S11    |                |
| 23      | 11/15/2007 | Code Generators            | Chapter 9 S12    | Out(6)         |
| 24      | 11/20/2007 | Thanksgiving Break         |                  |                |
| 25      | 11/22/2007 | Thanksgiving Break         |                  |                |
| 26      | 11/27/2007 | Optimizing Compilers       | Chapter 10       |                |
| 27      | 11/29/2007 | Optimizing Compilers II    | Chapter 10       | Due(5)         |
| 28      | 12/04/2007 | Security and Compilers     |                  |                |
| 29      | 12/06/2007 | Security and Compilers II  |                  | Due(6)         |
| Final   |            |                            |                  |                |

Table 1: Tentative Schedule