

SYLLABUS – Seattle Pacific University – Winter 2005

CSC 4310 – Compiler Design

CATALOG DESCRIPTION

CSC 4310 COMPILER DESIGN (4)

Studies programming language translation and compiler design concepts; language recognition, symbol table management, semantic analysis and code generation.

PREREQUISITES

Prerequisites: CSC 3310 and CSC 3750 or CSC/CPE/EE3760.

MEETING DATES, TIME and PLACE

Class meets Tuesday and Thursday Evenings 6:00pm – 7:20pm, OMH-245

Attendance is expected. SPU Snow Closure Hotline: (206) 281-2800

Final Exam: Tuesday, March 15, 2005 6:00pm – 7:20pm, OMH-245.

INSTRUCTOR

Mike Tindall OMH-240 (206) 281-2945 mht@spu.edu <http://myhome.spu.edu/mht>

Office Hours: T/Th 4:00 - 5:30pm (or by appointment).

I read my Email several times each day Monday through Friday and occasionally at other times.
If you have questions, this is the best way to reach me.

REQUIRED TEXT

Programming Language Pragmatics. Michael Scott, 2000. Academic Press. ISBN 1-55860-442-1.

Useful and Recommended Reference:

Lex & Yacc, Levine, Mason and Brown, 1992. O'Reilly. ISBN 1-56592-000-7.

COURSE OBJECTIVES and TOPICS

Objectives:

- to introduce the major concept areas of language translation and compiler design.
- to develop an awareness of the function and complexity of modern compilers.
- to provide practical, hands-on experience in compiler design, writing and modification.

Topics:

- Lexical Analysis, regular grammars, scanners.
- Syntactic Analysis, BNF notation, parsing theory.
- LL(k) and LR(k) parsers. Grammar transformations for parser compliance.
- Parser generators, translator writing systems.
- Semantic Analysis, symbol tables, intermediate text forms.
- Runtime machine models. Native code. Virtual machines. P-Code.
- Overview of Code Generation techniques, non-optimized.
Register management, storage allocation.

COURSE WORK and GRADING

The course assigned work will primarily involve a set of lab exercises to familiarize with the Flex/Bison tools and develop a small compiler for a simple programming language PL4310.

- Grading: (a) Programming Labs and Homework (70%)
 (b) Midterm Exam, Final Exam, Quizzes (30%)
 The student must pass the Final exam to pass the course.

Cygwin Tools

From <http://www.cygwin.com> , download the **Setup.exe** program and run it to download and install a (partial) toolset that includes at least the base utilities and the Development tools. Be sure your toolset includes Bison and Flex. A typical Cygwin install uses a home directory, e.g., **C:\Cygwin**. Once installed, you can launch the provided **Cygwin.bat** program to start up a Linux-lookalike Bash command window. Alternatively, you can simply add **C:\Cygwin\bin** to your system path (e.g., ControlPanel/System/Advanced/Environment), and then the Cygwin tools should run directly from a normal Windows cmd.exe window.

INDIVIDUAL WORK ONLY

All work is individual effort only, unless specific exceptions are made by the instructor. You may talk with other class members about programming style, understanding the problem, and debugging techniques. It is wrong to copy someone else's code or to ask them to debug your program. Violations may result in a zero for the first offense and failure of the course for the second.

STUDENTS WITH DISABILITIES

Please contact Disabled Student Services in the Center for Learning to request academic accommodations. Disabled Student Services sends Disability Verification Letters to your professors indicating the appropriate accommodations for the classroom based on your disability.

TEST DATES and POLICIES

The **Final Exam** (as published in the SPU time-schedule) is scheduled for:

Final Exam - Tuesday, March 15, 2005 6:00pm - 7:20pm OMH-245

The Final Exam will be closed-book and comprehensive.

PROFESSIONAL CODE OF ETHICS

From the ACM (Association for Computing Machinery)

<http://www.acm.org/constitution/code.html>

From the IEEE (Institute of Electrical and Electronics Engineers)

http://www.ieee.org/portal/index.jsp?pageID=corp_level1&path=about/whatis&file=code.xml&xsl=generic.xsl

Computer Science Department – Goals and Learning Objectives

The Computer Science department has identified several goals for students enrolled in our programs. Each of the courses in our curriculum are designed to help students achieve specific learning objectives that will help them progress towards these goals. As noted below (\Rightarrow), several of the Computer Science departmental learning objectives are particularly relevant to your studies in CSC 3310:

Goal 1: Help students develop problem-solving skills, especially those required to analyze, design and implement solutions involving the use of a computer.

- \Rightarrow **Objective 1:** Successful students will acquire the up-to-date technical knowledge and develop the skills needed for a successful start to careers in the computing industry.
- \Rightarrow **Objective 2:** Successful students will be able to develop solutions to problems that are new to them, and implement these solutions efficiently.
- \Rightarrow **Objective 3:** Successful students will be able to implement solutions utilizing different computer platforms and programming languages.
- Objective 4:** Successful students will develop the skills needed to work in small groups on medium to large scale projects.
- Objective 5:** Successful students will develop the ability to write technical documents that include specification, design, and implementation of major projects.
- Objective 6:** Successful students will be able to effectively disseminate information and results using both oral and written communication.

Goal 2: Provide a background in modern computing systems and the theoretical aspects of computer science.

- \Rightarrow **Objective 1:** Successful students will acquire the computer science knowledge required for graduate studies.
- \Rightarrow **Objective 2:** Successful students will understand the architecture, organization and programming of modern computing systems.
- \Rightarrow **Objective 3:** Successful students will understand the mathematical foundations of computer science, algorithm efficiency and computational complexity.

Goal 3: Challenge students to consider the ethical and social impacts of technology, enabling them to take responsible action informed by a Christian world view.

- Objective 1:** Successful students will be aware of ethical and social issues related to technology and recognize their impact.
- Objective 2:** Successful students will be able to evaluate potential ethical dilemmas and apply decision-making techniques to resolve them.

Goal 4: Prepare students for continued learning in a rapidly changing discipline.

- \Rightarrow **Objective 1:** Successful students will be aware of the rapid rate of change of technology and methodologies in computer science.
- \Rightarrow **Objective 2:** Successful students will be familiar with ways to gain knowledge and understanding of new developments in computer science and technology.
- Objective 3:** Successful students will be aware of alternatives for continuing education in computer science.

“As a community of learners, Seattle Pacific University seeks to educate and prepare students for service and leadership. We are committed to evangelical Christian faith and values, and to excellence in teaching and scholarship for the intellectual, personal and spiritual growth of students.”

– Mission Statement Seattle Pacific University

“The Department of Computer Science is dedicated to educating and preparing students for a variety of careers in business, scientific and engineering computing. We seek to provide a broad program of studies in theoretical and applied computer science informed by a Christian world view, graduating students who are equipped for continued professional development and service.”

– Mission of CSC Dept.