Course Number: 3460:440/540
Course Name: Compiler Design
Course Credits: 3
Schedule: Spring
Syllabus Date: June 9 2008
Prepared By: Tim O’Neil

Prerequisites:
Completion of 306 and 316, or 501, or equivalents with grades of C- or better.

Text:

Bulletin Description:
Techniques used in writing and modifying compilers including translation, loading, execution, symbol tables and storage allocation; compilation of simple expressions and statements. Organization of a compiler for handling lexical scan, syntax scan, object code generation, error diagnostics and code optimization. Use of compiler writing languages and boot-strapping. The course requires a project involving compiler writing.

Detailed Description:
An introduction to the fundamental techniques used in compiler construction.

Course Goals:
This course is designed to introduce the theory and practice of compiler construction. A major component of the course will involve the design and implementation of a compiler, providing experience for large scale software development.

Topics:
1. Introduction and overview
2. Lexical analysis: regular expressions, formal grammars and finite automata
3. Syntax analysis: top-down, bottom-up, and shift-reduce parsing
4. Semantic analysis: syntax-directed translation and type checking
5. Run-time environment: activation records and scoping, storage allocation, parameter passing, memory management
6. Intermediate code generation: symbol tables and intermediate representations
7. Code generation and optimization

Computer Usage:
Programming assignments will be made on a regular basis.

References: