

CS 460 - Course Syllabus – Fall 2008

Catalog Description

Lecture, 4 hours. A survey of the syntactic, semantic and implementation features of functional, procedural, object-oriented, logic and concurrent programming languages

Prerequisites

CS 252 (CS 250) and CS 315 (CS 254) or instructor consent. Knowledge of C/C++ and data structures are required. Students lacking background in C/C++ will need to familiarize themselves with the language within the first two weeks of the course. Students who have not acquired a strong foundation in the fundamentals of programming and data structures (i.e., earned a grade below a C- in CS 252 (CS 250) and CS 315 (CS 254)) are required to retake the course, or its equivalent, before taking this course.

Instructor

Dr. Tia Watts Darwin 116E (707) 644-2807
e-mail: tiawatts@cs.sonoma.edu (Message subject *MUST* begin with **CS 460**)
Office Hours: Tuesday 1:00 – 2:00 pm Office or Laboratory
Office Hours: Thursday 2:00 – 3:00 pm Office or Laboratory

Class Meetings

Lecture: Tuesday/Thursday 10:00-11:50 am Darwin 29

Course Objectives

Upon successful completion of this course, the student will be able to:

- Describe the steps and concepts required to evaluate a new programming language.
- Describe and implement the steps required to learn a new programming language.
- Describe the highlights of the development of programming languages.
- Describe and differentiate between functional, procedural, object-oriented, and logic programming paradigms.
- Describe the common features of programming languages.
- Describe the implementations of a variety of programming language features.
- Design and implement a recursive descent language translator.

In addition, the student will have used (but not mastered) a variety of programming languages representing different programming paradigms.

Important Dates

26 August 2008	First day of classes
9 September 2008	Last day to ADD or DROP courses
22 September 2008	Last day to WITHDRAW from courses online
16 October 2008	Midterm Exam
11 November 2008	Veteran's Day Observed (No classes – University closed)
26-28 November 2008	Thanksgiving Break (No classes – University closed)
12 December 2008	Last day of classes
16 December 2008	Final Exam: 11:00 am – 12:50 pm

Course Materials

Text Concepts of Programming Languages (8th edition) by Robert W. Sebesta
Addison Wesley Publishing Company, 2008.

Other Materials

CS 460 Course Manual
Loose Leaf Binder; Hole Punch; Small Stapler; Folder

Coursework

Lecture: The proposed outline of the topics to be covered appears in the course schedule. Students are expected to attend all lectures and to read the relevant sections of the text prior to lecture. Students are responsible for making up the missed work if they are absent.

Tutorials: The tutorials for this course are designed to give the student an opportunity to work with programming languages from a variety of programming paradigms. Tutorial exercises must be submitted by the dates indicated in the course schedule.

Programming Projects: A multi-stage programming project will be assigned during the semester. Each stage will require that you write one or more C program modules. Stage due dates and times will be stated on the project assignment handout. Programs must be submitted by the due date and time based on the UNIX date/time stamp on the file(s). No projects will be accepted after the due date and time.

Midterm and Final Exam: The examinations are based on the assumption that each student is responsible for knowing the material covered in class, in the relevant sections of the course text, and the tutorials in the course manual. All exams are closed book; however, one handwritten 8 ½ by 11 sheet of notes (front and back) will be allowed. Exams will include questions related to the programming languages covered in the tutorials. The final exam will be comprehensive. There may be one or more unannounced quizzes given during the semester. Exams cannot be made up unless you have made arrangements with the instructor prior to the date of the exam. Quizzes cannot be made up.

Grading

Homework and Tutorial Exercises	30%
Programming Project	20%
Exams and Quizzes	50%

Grading Scale:

100...	93...	90...	87...	83...	80...	77...	73...	70...	67...	63...	60...	0
A	A-	B+	B	B-	C+	C	C-	D+	D	D-	F	

Note: You must separately earn a passing grade on the tutorial exercises, programming projects, and the exams in order to pass the course.

CS Majors must take this course for a letter grade. University guidelines regarding the grade of Incomplete will be strictly adhered to. Incomplete grades will only be given for circumstances beyond a student's control; inability to keep up with the work due to an excessive course load, for example, is insufficient to warrant an Incomplete. The University also requires that, to be a candidate for an Incomplete grade, the student must currently be doing C (or better) work in the course.