

Course: CS-481 OBJECT-ORIENTED PROGRAMMING (Section 1)

Time & Loc: Lecture: M, Tu, Th: 11:00am - 11:50am (L-309)

Instructor: Prof. Jeff Blessing

E-mail: blessing@msoe.edu

Office: S-336

Hours: M, Tu, Th: 10:00am - 10:50am, and by appointment

Phone: 277-7194

Text: The C++ Programming Language, by Bjarne Stroustrup, Addison-Wesley, 1997.

www: A web page which contains printed copies of all the slides used in this course (.pdf format) can be obtained from www.msoe.edu/~blessing/cs481.html
Please check the course page regularly, since assignments and any changes to the course material will be posted to the web page throughout the quarter.

Objective: The goal of the course is to teach an experienced C language programmer the object-oriented features which C++ brings to the unified C/C++ language. (To do this, several programs will be assigned to students, which must be done on an individual basis. Also, to reinforce the basic concepts in a timely manner, weekly quizzes will be given, in addition to a midterm and final exam.)

<i>Grading:</i>	Programs & Weekly Quizzes	50%
	Midterm (week 6)	20%
	Final (week 11)	30%

Attendance: Although not reflected for the grading this course, attendance is strongly encouraged. If chronic absenteeism is a problem, attendance will be formally taken. Exam material will come from both the text and class discussions.

Late Policy: Assignments are due at the beginning of the class period that was designated as their due date. A 10% penalty will be assessed for each day an assignment is late.

Programs: Each student will individually write 5 or 6 programs in C++ during the quarter. While it is beneficial for students to learn from each other, as well as in class and from written material, it is imperative that each student does their own work on the programs, since they will form the basis of most of the quiz and exam content.

Prerequisites: The following topics form the basic background which the student must have in order to perform the required work for the course:

1. A *thorough* understanding of all C language constructs including structures, pointers and other procedural programming practices.
2. A *fundamental* understanding of data structures (arrays, strings, lists, trees, graphs, etc.) and the algorithms that manipulate them.
3. A *basic* understanding of the Software Engineering process, including the analysis, specification, design, implementation, and testing of software systems.
4. Access to (and knowledge of using) browsers on the World Wide Web.