

COURSE DELETED FROM SSU CATALOG AND COMPUTER SCIENCE MAJOR
Last offering of this course was Fall 2016

CSC 273 Advanced Applications of C

4 cr.

Instructor: TBA
email: TBA@salemstate.edu

Office: location
Office Hours: days and times

Phone: (978) 542-extension

Section	Time	Room	Final Exam
nn	days and times	location	date and time
Lnn	days and times	location	

Catalog description:

This course presents the C programming language, with emphasis on its particular strengths and special features, including the rich variety of data types and operators, pointers, and bit manipulation. Programming assignments will highlight applications for which the special capabilities of C are particularly well suited, such as systems programming, text processing, computer graphics, and real-time control. Three lecture hours and three hours of scheduled laboratory per week, plus extensive programming work outside of class.

Prerequisite: CSC 115 or CSC 202J.

Goals:

- CG01: to enhance students' skills in problem analysis and program design and implementation through the presentation of complex applications;
- CG02: to present a thorough coverage of the C programming language, with emphasis on its special features.

Objectives:

Upon completion of this course the student will have demonstrated the ability to:

- CO01: understand and utilize the syntax and special capabilities of the C language, including the preprocessor, header files, pointers, operators, side effects, binary files, and low-level features such as bit manipulation;
- CO02: select (or create) appropriate algorithms, data structures, and language features for the solution of a complex problem, and use these ingredients effectively to obtain a solution to the problem;
- CO03: employ a clear, consistent, readable style in the implementation of a problem solution;
- CO04: produce clear documentation for a problem and its solution.

Program Outcome vs. Course Objectives matrix

Program Objective (condensed form)	CO01	CO02	CO03	CO04
PO-A: apply knowledge of computing and math	✓	✓	✓	✓
PO-B: analyze a problem and define its computing requirements		✓		✓
PO-C: design, implement and evaluate applications		✓	✓	✓
PO-D: function effectively in teams to accomplish a common goal			✓	✓
PO-E: professional, ethical, and social responsibilities				
PO-F: communicate effectively with a range of audiences				
PO-G: local and global impact of computing on people and society				

Program Objective (condensed form)	CO01	CO02	CO03	CO04
PO-H: need for continuing professional development				
PO-I: use current techniques, skills, and tools	✓	✓	✓	✓
PO-J: apply theory and principles to model and design systems		✓		
PO-K: apply design and development principles in constructing software		✓		
note - full statements of the Program Outcomes (program objectives) for the Computer Science Major can be found in the document <i>Computer Science Major Program Educational Objectives and Program Outcomes</i> on the Assessment page of the Computer Science Major (cs.salemstate.edu)				

Topics:

- Programming language concepts
 - programming paradigms (two hours)
 - procedural
 - object-oriented
 - functional
 - declarative
 - multi-paradigm
 - emerging and/or specialized paradigms
 - type systems (one hour)
 - "strongly-typed" vs "weakly-typed" vs. "type safe"
 - static (compile-time) type checking
 - dynamic (run-time) type checking
 - memory allocation and management (one hour)
 - static vs. dynamic
 - direct vs. indirect
- review of standards for algorithm design and programming style
- similarities and differences between C and other procedural languages (such as Ada) and object-oriented languages (such as C++ and Java)
- the C programming environment:
 - the standard C library
 - header files
 - the C preprocessor and preprocessor directives (**#include**, **#define**, **#ifdef**, etc.)
 - separate compilation and linking
 - the run-time environment---the program text, data, stack, and heap areas
- C language syntax and features: **PF1(3), PF3(10), PL4(2), PL5(2), AR(2)**
 - the data type hierarchy
 - identifiers, constants, variables
 - declaration and initialization
 - type conversions (explicit and implicit)
 - operators: arithmetic, relational, logical, increment and decrement, bitwise
 - functions and parameter passing; function declarations (prototypes)
 - structure of a C program
 - scope rules
 - control flow: loops, conditional statements and branching
 - input/output:
 - files and streams
 - the FILE type
 - text I/O, interactive I/O, binary I/O, direct access I/O
 - addresses and pointers
 - arrays
 - strings
 - structures, unions
- efficiency considerations:
 - choosing the most appropriate C implementation for the task at hand

- efficiency vs. readability
- applications (topics chosen from the following):
 - string processing
 - searching and sorting
 - data compression
 - data encryption
 - numeric applications (including high-precision arithmetic)
 - graph algorithms
 - simulations

**PF2(3), PF4(2), PF5(1), SE1(7), SE6(1)
AL2(3), AL3(3)**

The emphasis of the course is on effective use of the C language in conjunction with standard software engineering principles, including portability and reusability of code. Specific methods and standards for problem specification, algorithm design, coding style, testing, and maintenance will be employed. Some features of the C++ language may be introduced and contrasted with the corresponding C versions.

Laboratory exercises will concentrate on language syntax, control structures, and standard C idioms for common programming tasks. In addition, there will be four to six substantial programming assignments. Some of these will build on earlier problem solutions. Assignments may be drawn from the areas listed above and from other areas of specific interest to the instructor and/or the class.

All programs must conform to departmental guidelines for algorithm design and implementation. Laboratory reports must conform to the written guidelines supplied by the instructor. Regardless of numeric average or individual grades on assignments or examinations, a student will NOT be eligible for a passing grade in the course unless he or she has submitted a laboratory report for every programming assignment within the time-frame specified by the instructor.

The course grade will be determined using the following approximate weights: laboratory exercises - 20%; programming assignments – 40%; final examination - 20%; other tests and written homework - 20%.

Course Objective / Assessment Mechanism matrix

	Tests / Written Homework	Lab Exercises	Programming Projects	Final Examination
CO01	✓	✓	✓	✓
CO02			✓	✓
CO03	✓	✓	✓	
CO04	✓		✓	

Bibliography:

Bates, Martin P. **Programming 8-bit PIC Microcontrollers in C: With Interactive Hardware Simulation.** Newnes, 2008.
 Bronson, Gary. **A First Book of ANSI C. Fourth Edition.** Cengage Learning, 2006.
 Collopy, David M. **Introduction to C Programming: A Modular Approach.** Prentice Hall, 2002.
 Deitel, Paul J.; Deitel, Harvey M. **C: How to Program. Sixth Edition.** Prentice Hall, 2010.
 Kernighan, Brian W.; Ritchie, Dennis M. **The C Programming Language. Second Edition.** Prentice Hall, 1988.
 Koffman, Elliot B.; Hanly, Jeri R. **Problem Solving and Program Design in C.** Addison-Wesley, 2009.
 Loudon, Kyle. **Mastering Algorithms with C.** O'Reilly, 1999.
 Prata, Stephen **C Primer Plus, 5th ed., Sams, 2005.**
 Schildt, Herbert. **C/C++ Programmer's Reference. Second Edition.** McGraw-Hill, 2000.
 Schildt, Herbert. **C: The Complete Reference. Fourth Edition.** Osborne/McGraw-Hill, 2000.
 Seacord, Robert C. **Secure Coding in C and C++.** Addison-Wesley, 2006.
 Sedgewick, Robert. **Algorithms in C. Third Edition. Parts 1-4: Fundamentals, Data Structures, Sorting, Searching. Third Edition.** Addison-Wesley, 1998.
 Sedgewick, Robert. **Algorithms in C, Part 5: Graph Algorithms. Third Edition.** Addison-Wesley, 2002.

Academic Integrity Statement:

“Salem State University assumes that all students come to the University with serious educational intent and expects them to be mature, responsible individuals who will exhibit high standards of honesty and personal conduct in their academic life. All

forms of academic dishonesty are considered to be serious offences against the University community. The University will apply sanctions when student conduct interferes with the University primary responsibility of ensuring its educational objectives." Consult the University catalog for further details on Academic Integrity Regulations and, in particular, the University definition of academic dishonesty.

The Academic Integrity Policy and Regulations can be found in the University Catalog and on the University website ([http://catalog.salemstate.edu/content.php?catoid=13&navoid=1295#Academic Integrity](http://catalog.salemstate.edu/content.php?catoid=13&navoid=1295#Academic_Integrity)). The formal regulations are extensive and detailed - familiarize yourself with them if you have not previously done so. A concise summary of and direct quote from the regulations: "Materials (written or otherwise) submitted to fulfill academic requirements must represent a student's own efforts". *Submission of other's work as one's own without proper attribution is in direct violation of the University's Policy and will be dealt with according to the University's formal Procedures. Copying without attribution is considered cheating in an academic environment - simply put, **do not do it!***

University-Declared Critical Emergency Statement:

In the event of a university-declared emergency, Salem State University reserves the right to alter this course plan. Students should refer to www.salemstate.edu for further information and updates. The course attendance policy stays in effect until there is a university-declared critical emergency.

In the event of an emergency, please refer to the alternative educational plans for this course, which will be distributed via standing class communication protocols. Students should review the plans and act accordingly. Any required material that may be necessary will have been previously distributed to students electronically or will be made available as needed via email and/or Internet access.

Equal Access Statement:

"Salem State University is committed to providing equal access to the educational experience for all students in compliance with Section 504 of The Rehabilitation Act and The Americans with Disabilities Act and to providing all reasonable academic accommodations, aids and adjustments. **Any student who has a documented disability requiring an accommodation, aid or adjustment should speak with the instructor immediately.** Students with Disabilities who have not previously done so should provide documentation to and schedule an appointment with the Office for Students with Disabilities and obtain appropriate services."

<p>Note: This syllabus represents the intended structure of the course for the semester. If changes are necessary, students will be notified in writing and via email.</p>
