

CSE 5042 (Proposed): Systems I: Introduction to Low-Level Programming and Computer Organization

Course Description

Introduction to computer architecture at machine and assembly language level; pointers and addressing; C programming at machine level; computer organization.

Prior Course Number: CSE 360 and CSE 459.21

Transcript Abbreviation: Sys I: Comput Org

Grading Plan: Letter Grade

Course Deliveries: Classroom

Course Levels: Undergrad, Graduate

Student Ranks: Sophomore, Junior, Senior, Masters, Doctoral

Course Offerings: Autumn, Spring, Summer

Flex Scheduled Course: Never

Course Frequency: Every Year

Course Length: 14 Week

Credits: 3.0

Repeatable: No

Time Distribution: 4.0 hr Lec

Expected out-of-class hours per week: 5.0

Graded Component: Lecture

Credit by Examination: No

Admission Condition: No

Off Campus: Never

Campus Locations: Columbus

Prerequisites and Co-requisites: 2122 (230) or 2123 (314) or 2231 or 321, and 2321 or Math 2366 or Math 366

Exclusions: Not open to students with credit for 360 or 2421.

Cross-Listings:

Course Rationale: To give non-majors the option to pick up pre-reqs in order to be prepared to take more advanced CSE courses.

The course is required for this unit's degrees, majors, and/or minors: No

The course is a GEC: No

The course is an elective (for this or other units) or is a service course for other units: Yes

Subject/CIP Code: 14.0901

Subsidy Level: Doctoral Course

Programs

Abbreviation	Description
BS CSE	BS Computer Science and Engineering

General Information

This course is appropriate for non-CSE majors only.

Course Goals

To master programming with pointers in C
To be competent with application development and debugging in Unix environments
To be competent in programming with dynamic data structures in C, and in using C string and I/O features, bit operations, and function pointers
To be familiar with overall organization and design of computer systems
To be competent with representation and manipulation of information in computer systems
To be familiar with machine encoding of instructions, and be competent with a particular real or hypothetical instruction set
To be familiar with programming in assembly language
To be familiar with Linking (static linking, relocatable object files, symbols and symbol tables, symbol resolution, relocation, loading executable object files)

Course Topics

Topic	Lec	Rec	Lab	Cli	IS	Sem	FE	Wor
Transitioning from Java to C, Basic C syntax, working in Unix Environments	6.0		2.0					
C pointers and memory allocation/deallocation. Programming dynamic data structures with C (linked lists, arrays, including multi-dimensional arrays accessed through pointers, trees), string manipulation, pointer casting, null/void pointers.	6.0		2.0					
Other misc C features: I/O operations, bit operations, function pointers, command line argument passing	3.0		1.0					
Debugging in Unix with gdb/xgdb, Use of Makefile, Other Unix features	3.0		2.0					
Introduction to Computer Systems Organization	3.0							
Representation and manipulation of information (information storage, integer representation, integer arithmetic, floating point)	6.0		2.0					
Machine level representation of programs (program encoding, data formats, accessing information, arithmetic and logical operations, control, procedures, array allocation and access, alignment)	9.0		2.0					
Programming with an assembly language: simple use of registers and arithmetic operations, conditionals and loops, accessing arrays in assembly, procedure calls in assembly.	3.0		3.0					
Linking (static linking, relocatable object files, symbols and symbol tables, symbol resolution, relocation, loading executable object files)	3.0							

Representative Assignments

Introductory C programming (for students with 2 semesters of advanced programming)
Dynamic data structure based programming in C
Use of command line argument passing and function pointers, advanced string/buffer manipulation
Using bit operations in C to simulate hardware computer arithmetic algorithms
Introductory assembly programming

Grades

Aspect	Percent
Programming Assignments (~5)	15%
Written Assignments (~3)	12%
In-Class Assignments	8%
Mid-term 1	15%
Midterm 2	20%
Final Exam	30%

Representative Textbooks and Other Course Materials

Title	Author
<i>Computer Systems: A Programmer's Perspective</i>	Bryant and O'Hallaron
<i>Pointers with C</i>	Kenneth Reek

ABET-EAC Criterion 3 Outcomes

Course Contribution		College Outcome
*	a	An ability to apply knowledge of mathematics, science, and engineering.
*	b	An ability to design and conduct experiments, as well as to analyze and interpret data.
**	c	An ability to design a system, component, or process to meet desired needs.
	d	An ability to function on multi-disciplinary teams.
***	e	An ability to identify, formulate, and solve engineering problems.
	f	An understanding of professional and ethical responsibility.
	g	An ability to communicate effectively.
	h	The broad education necessary to understand the impact of engineering solutions in a global and societal context.
	i	A recognition of the need for, and an ability to engage in life-long learning.
	j	A knowledge of contemporary issues.
**	k	An ability to use the techniques, skills, and modern engineering tools necessary for engineering practice.

BS CSE Program Outcomes

Course Contribution		Program Outcome
**	a	an ability to apply knowledge of computing, mathematics including discrete mathematics as well as probability and statistics, science, and engineering;
**	b	an ability to design and conduct experiments, as well as to analyze and interpret data;
***	c	an ability to design, implement, and evaluate a software or a software/hardware system, component, or process to meet desired needs within realistic constraints such as memory, runtime efficiency, as well as appropriate constraints related to economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability considerations;
	d	an ability to function on multi-disciplinary teams;
**	e	an ability to identify, formulate, and solve engineering problems;
	f	an understanding of professional, ethical, legal, security and social issues and responsibilities;
	g	an ability to communicate effectively with a range of audiences;
	h	an ability to analyze the local and global impact of computing on individuals, organizations, and society;

Course Contribution		Program Outcome
	i	a recognition of the need for, and an ability to engage in life-long learning and continuing professional development;
	j	a knowledge of contemporary issues;
***	k	an ability to use the techniques, skills, and modern engineering tools necessary for practice as a CSE professional;
***	l	an ability to analyze a problem, and identify and define the computing requirements appropriate to its solution;
***	m	an ability to apply mathematical foundations, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices;
**	n	an ability to apply design and development principles in the construction of software systems of varying complexity.

Prepared by: Kathryn Reeves