
COMPUTER SCIENCE

Chair: Robert Chen

Professor: Suban Krishnamoorthy

Associate Professor: Robert Chen

Assistant Professors: Cornelia Breuning, Jeff Gao, *David Keil

*Sabbatical: Spring 2005

COMPUTER SCIENCE MAJOR

The Department offers courses leading to a Bachelor of Science in Computer Science with concentrations in Software Engineering and Information Systems, and a minor in Computer Science. The goal of these programs is to train graduates for successful careers in industry and to prepare students for graduate studies. Students are encouraged to do cooperative work with local companies for up to one year, receiving up to four course credits, job experience, and (normally) compensation in the process. Such experience is also beneficial in seeking post-graduate employment.

The General Education Requirement:

All students must satisfy a general education requirement consisting of twelve (12) courses. See page 34 of this catalog.

Course Prerequisites

Courses may have specified conditions for enrollment, such as prior completion of less advanced courses, permission of the instructor, or appropriate placement test scores. Students should refer to course descriptions in the department listings for prerequisite requirements.

CONCENTRATION IN SOFTWARE ENGINEERING (CSS)

The software engineering concentration prepares students for careers in software development.

Required Computer Science courses (11):

- 63.152 Computer Science I
- 63.252 Computer Science II
- 63.258 Introduction to Operating Systems Using UNIX™

- 63.266 Digital Logic, Organization, and Assembly Language
- 63.271 Data Structures
- 63.355 Computer Architecture
- 63.362 Software Engineering
- 63.347 Analysis of Algorithms
- 63.460 Theory of Computing
- 63.465 Operating Systems Internals
- 63.477 Computer Networking

Two (2) Computer Science electives from the following:

- 63.320 Windows Server & Client Management
- 63.340 UNIX™ System Administration
- 63.360 Database Management
- 63.373 Advanced Web Technologies
- 63.404 Seminar in Computer Science
- 63.455 Human-Computer Interaction
- 63.428 Software Quality Assurance

Four (4) required Mathematics courses

- 43.117 Introduction to Statistics (Goal 2)
- 43.219 Calculus I (Goal 2)
- 43.292 Discrete Mathematics I
- 43.220 Calculus II

Two (2) science courses. Software Engineering majors must meet a two-semester laboratory science requirement. The following course sequences qualify:

- 53.211 Principles of Physics I (Goal 6) **and**
53.212 Principles of Physics II
- or**
- 33.107 Principles of Chemistry (Goal 6) **and**
33.108 Principles of Chemistry and Quantitative Analysis (Goal 6)
- or**
- 73.231 Physical Geology (Goal 6) **and** 73.232
Historical Geology (Goal 6)

CONCENTRATION IN INFORMATION SYSTEMS (CSI)

This concentration trains students for technical careers in information systems such as database administration, network administration, systems analysis and design, and software quality assurance.

Required Computer Science courses (8):

- 63.108 HTML, JavaScript Programming, and Web Site Development
- 63.120 Introduction to Information Technology

- 63.152 Computer Science I
 63.252 Computer Science II
 63.258 Introduction to Operating Systems Using UNIX™
 63.360 Database Management
 63.373 Advanced Web Technologies
 63.376 Networking Technologies

Three (3) required Mathematics courses:

- 43.117 Introduction to Statistics (Goal 2)
 43.219 Calculus I (Goal 2)
 43.292 Discrete Mathematics I

Five (5) required Business Courses:

- 12.101 Principles of Macroeconomics (Goal 10) **or**
 12.102 Principles of Microeconomics (Goal 9)
 12.121 Introduction to Managerial Accounting
 12.280 Applied Organizational Theory and Management
 12.385 Business Systems Analysis and Design
 12.477 Management Issues in Information Technology

Three (3) courses from at least two of the following categories.

Network and System Administration

- 63.320 Windows Server and Client Management
 63.340 UNIX™ System Administration

General Information Systems Electives

- 63.205 Introduction to Visual Basic
 63.266 Digital Logic, Organization, and Assembly Language
 63.404 Seminar in Computer Science
 63.428 Software Quality Assurance
 63.455 Human-Computer Interaction
 63.490 Directed Study

Computer Graphics and Technical Writing

- 21.286 Professional Writing
 21.471 Business Writing
 21.472 Technical Writing
 84.140 Introduction to the Internet, Graphics and Multimedia

Note: Students may need to take additional math courses if their math placement score does not qualify them for Calculus. These would be taken as free electives.

MINOR IN COMPUTER SCIENCE (5 courses)

Required Courses (4)

- 63.120 Introduction to Information Technology
 63.152 Computer Science I
 63.252 Computer Science II
 63.258 Introduction to Operating Systems Using UNIX™

Elective (1)

One other course from the Computer Science Department.

MINOR IN INFORMATION TECHNOLOGY (see page 31)

COURSE DESCRIPTIONS

63.108 HTML, JavaScript Programming, and Web Site Development

An introduction to various features of HTML and JavaScript languages such as tags, images, buttons, forms, and animation files. Students learn to program using these languages to create interactive Web pages. Students create a complete Web site using various tools including a browser and image editor.

63.120 Introduction to Information Technology

An overview of computer concepts and Information Technology (IT). Applications of IT in various disciplines

are illustrated and the role of information technology in contemporary society, including issues of intellectual property, ethics, privacy and security is discussed. Students are introduced to windowed environments, file management, problem-solving tools and a high-level programming language. Advanced concepts of word processing, spreadsheets, and presentation software required for effective communication, analysis, and design are explored. **Note:** Students cannot receive credit for both this course and 63.151 Personal Computer (PC) Fundamentals and Applications. Students in the IT minor who have taken 63.151

Personal Computer (PC) Fundamentals and Applications should take 63.108 HTML, JavaScript Programming and Web Site Development in lieu of 63.120 Introduction to Information Technology.

63.151 Personal Computer (PC) Fundamentals and Applications

An overview of computers and how they are used both in academic and professional environments. Basic to advanced applications of word processing, spreadsheet and presentation software are used for problem solving, data analysis and effective communication. Issues including pri-

vacancy, piracy, ethics and the impact of current computer trends on Information Technology are also discussed.

Note: Students cannot receive credit for both 63.120 Introduction to Information Technology and 63.151.

63.152 Computer Science I

An introduction to problem solving using the Java programming language. The course stresses algorithms, object-oriented programming in graphical environments, documentation, testing, and debugging. Topics include hardware basics and number systems, classes, methods, control structures, types, virtual-machine concepts, Internet and client-server computing, human-computer interaction, social, professional, and ethical issues, and general features of programming languages.

Corequisite: 43.123 College Algebra **or** minimum score of 2 on the math placement examination.

63.205 Introduction to Visual BASIC

An introduction to the Visual BASIC programming environment and the event-driven programming model. Topics include forms and controls, code construction, functions and parameter passing, use of lists and combo box controls, creation of menus, program structure, error handling, and debugging of programs.

Prerequisite: 63.152 Computer Science I, **or** familiarity with a Windows environment and programming experience in any language.

63.252 Computer Science II

An intermediate programming course that emphasizes debugging, documentation, and modular and object-oriented design with tools such as the Unified Modeling Language. Topics include event-driven programming, string and array manipulation, sorting and searching, file operations, dynamic memory allocation, inheritance, polymorphism, and exception handling.

Prerequisites: 43.200 Precalculus (may be taken concurrently) and 63.152 Computer Science I.

63.258 Introduction to Operating Systems Using UNIX™

An introduction to the basics of networking and operating systems. Topics include the evolution and overview of operating systems, operating system principles, interfaces (GUI, command line, and API), command processors and utilities, file systems, access control, processes, programming and scripting, user accounts and authentication, OSI model, Internet Protocol, networking utilities, net computing, client-server model, FTP, and network file systems. The UNIX™ operating system is used as a primary reference for illustration and hands-on experience.

63.259 Object-Oriented Programming Using C++

A presentation of advanced features of programming with an emphasis on class construction and object-oriented design and programming. Topics covered include namespaces, encapsulation, function and operator overloading, templates, collection classes, containers, iterators, inheritance, polymorphism, and exception handling. Course touches on Standard Template Libraries (STL).

Prerequisite: 63.252 Computer Science II.

63.261 Computer Organization and Assembly Language Programming

Introduction to computer organization and assembly language programming. Representation of data, addressing modes, CPU instructions, and pseudo-instructions. Discussion of I/O and interrupt programming.

Prerequisite: 63.252 Computer Science II.

63.266 Digital Logic, Organization, and Assembly Language

An introduction to digital logic components and their organization in computer systems. Topics include the realization of logic components such as decoders, ALUs, registers and how they are combined to form CPUs and controllers. The course discusses the software interface, including machine instructions, the interrupt system, and sample I/O interfaces. The course also

covers assembly language and the organization of low-level software to control the computer, including representation of data, addressing modes, CPU instructions, and pseudo-instructions. **Note:** Students who have taken both 63.261 Computer Organization and Assembly Language Programming **and** 63.321 Digital Electronics cannot receive credit for 63.266 Digital Logic, Organization, and Assembly Language.

Prerequisite: 63.252 Computer Science II

63.271 Data Structures

An in-depth presentation of recursion, collections and iterators, fundamental techniques in graphics, and threading. Students implement linked lists, stacks, queues, trees, heaps, graphs, hash tables and related algorithms. Students implement a significant programming project.

Prerequisites: 43.200 Precalculus **and** 63.252 Computer Science II.

63.320 Windows Server & Client Management

Designed to teach students how to plan, set up, and configure Windows server and client workstations with TCP/IP enterprise protocol; monitor and control security within the systems network; administer user accounts, system disks, file system, mail, print and remote access services; and recover from server failures.

Prerequisite: 63.252 Computer Science II.

63.321 Digital Electronics

An introduction to the techniques of designing and constructing digital circuits with emphasis on understanding the logic circuits of a digital computer. Topics discussed include: application of Boolean algebra to logic design; operation of diodes and transistors; construction of IC logic gates; characteristics of TTL, ECL and CMOS logic families; using Karnaugh Maps to design combinatorial circuits; decoders; priority encoders; multiplexers; parity circuits; adders, subtractors and ALU's; latches and flipflops; counters; shift registers, and the design and analysis of synchronous state machines.

Includes a weekly three-hour laboratory period.

Prerequisite: 63.152 Computer Science I.

63.330 Java Programming

A comprehensive study of object-oriented programming using Java. Topics include objects, core Java classes, writing Applets and Applications, Servlets, thread control, networking, RMI, input-output streams, Java User Interface, Abstract Windows Toolkit, working with URLs, security, and error handling using exceptions.

Prerequisite: 63.259 Object-Oriented Programming Using C++.

63.340 UNIX™ System Administration

A comprehensive coverage of UNIX™ system administration and management. Topics include system setup, booting and shutdown, management of user accounts, file systems, disk management and serial device management, backup and restore, print and network configuration, security, system logs, performance monitoring, and maintenance.

Prerequisite: 63.258 Introduction to Operating Systems Using UNIX™.

63.347 Analysis of Algorithms

A presentation of asymptotic time and space complexity of sequential and parallel algorithms, using big-O and related notation. Complexity classes P and NP (tractable and intractable problems) and verification of algorithms by formal methods are also discussed.

Prerequisites: 63.271 Data Structures and 43.292 Discrete Mathematics I.

63.355 Computer Architecture

A study of the architecture of computer systems, the relationship between user applications and computer architecture as well as the relationships between operating systems, system software, and computer architecture. Topics covered: computer organization, user-oriented aspects of computer structure, register transfer operations, computer control with strong emphasis on micro-programming, and operating system level of computer

architecture. Advanced topics including multi-level computers, program portability, virtual machines, and high level language-oriented machines.

Prerequisites: 63.266 Digital Logic, Organization and Assembly Language and 63.271 Data Structures.

63.360 Database Management

A comprehensive coverage of database concepts, design, and implementation. Topics include systems analysis, data modeling, relational databases, logical design, normalization, user interfaces, query processing including SQL, database administration, security, backup and restore, and performance evaluation. Popular databases are referenced. **Note:** Students cannot receive credit for both 63.360 Database Management and 63.467 Data Base Management Systems.

Prerequisite: 63.258 Introduction to Operating Systems Using UNIX™.

63.362 Software Engineering

Principles of systematic program design and implementation. Structured programming and top-down segmented program development. Stepwise program development. Chief Programmer Team, HIPO, and other programming techniques. Decision tables. Program correctness and verification. Software reliability.

Prerequisite: 63.271 Data Structures.

63.373 Advanced Web Technologies

An introduction to dynamic database-driven Web site development. Students learn about client- and-server-side scripting, maintaining persistent information on the Web, and user-interface design concepts.

Prerequisites: 63.108 HTML, JavaScript Programming, and Web Site Development, 63.152 Computer Science I and 63.360 Database Management.

63.376 Networking Technologies

An introduction to net-centric computing, communication and networking, network security, network management, client-server computing, wireless and mobile computing, and

network applications. Various LAN and WAN networking technologies such as Ethernet, Token Ring, wireless network, virtual private network, personal area network, Storage Area Network, Internet, ATM, telecommunication networks, multimedia data technologies, and internetworking are covered. Business aspects of computer networks are also covered. **Note:** Students cannot receive credit for both 63.376 Networking Technologies and 63.476 Networking Technologies.

Prerequisite: 63.258 Introduction to Operating Systems Using UNIX™

63.404 Seminar in Computer Science

An exploration of an advanced topic in computer science. The particular topic is announced at least one semester in advance.

Prerequisite: Permission of the instructor.

63.428 Software Quality Assurance

A study of software quality assurance processes and test tools. Topics include requirements analysis, test specifications, test schedule development, installation and test tools, standalone product testing, systems integration testing, interoperability, problem reporting and tracking, product development and quality assurance processes, QA management, product life cycle, software metrics for quality, and product release.

Prerequisite: 63.258 Introduction to Operating Systems Using UNIX™

63.455 Human-Computer Interaction

A survey of the field of Human-Computer Interaction (HCI). Emphasis is placed on design strategies for creating and refining an effective user interface throughout the software product life cycle. Task analysis, approaches to user interface design, modeling, and user interface standards are covered as student groups develop a user interface for a software product.

Prerequisites: 63.152 Computer Science I and 63.360 Database Management.

63.460 Theory of Computing

An introduction to theoretical computer science and some key applications. Course examines models of computation, including finite automata, transducers, pushdown automata, and Turing machines. Concepts of formal language theory are applied to lexical analyzer and compiler construction in programming-language translation. The course will include an introduction to the notions of computability and computational complexity, concepts used in parallel computation, and some aspects of artificial intelligence.

Prerequisites: 43.292 Discrete Mathematics I **and** 63.271 Data Structures.

63.465 Operating Systems Internals

An in-depth coverage of operating system principles and internals such as processes, threads, CPU scheduling, concurrency and process synchronization, deadlock, inter-process communication, memory management, virtual memory, file systems, I/O systems, distributed systems, and protection and security. Several of the concepts are implemented through a programming project.

Prerequisites: 63.258 Introduction to Operating Systems Using UNIX™, 63.271 Data Structures, **and** 63.355 Computer Architecture.

63.477 Computer Networking

An introduction to the seven layers of the OSI model. General topics include network topologies; introduction to Local Area Networks, such as Ethernet, Token Ring, and FDDI; and internet-working. Special topics include TCP/IP and FTP protocols, socket interface, Remote Procedure Call (RPC), and client-server architecture.

Prerequisite: 63.271 Data Structures **or** equivalent computer industry experience including strong programming skills in C or C++.

63.481 Microsoft Windows Programming

A study of Windows programming and application development for experienced C++ programmers. Topics include: using Windows programming models, Windows development environments, Windows classes, objects, and messages; the Windows API; painting and text output; keyboard and mouse input; Windows timer and memory management; Windows resources, menus and dialogue boxes; and elements of graphics device interface. Additional topics are chosen from the following: clipboard usage, dynamic data exchange, object linking and embedding, multiple document interface, printer management, fonts, dynamic link libraries, application framework tools, and client/server applications.

Prerequisites: 63.261 Computer Organization and Assembly Language Programming and 63.271 Data Structures.

63.490 Directed Study

Under the direction of a faculty member, the student researches a topic or topics in computer science. Students should make arrangements with the faculty member who is to direct their work one semester in advance of the work.

63.495 Cooperative Program in Computer Science (credit-two courses)

A cooperative program in which students are placed in computer positions in corporate or government settings. Students are expected to work a minimum of 25 hours per week. These positions are normally compensated. Upon completion of the course, students will provide written documentation concerning their work projects.

Prerequisite: Upperclass standing in computer science with at least a 3.0 average in computer courses **or** permission of instructor.

63.496 Cooperative Program in Computer Science (credit-two courses)

A cooperative program in which students are placed in computer positions in corporate or government settings. Students are expected to work a minimum of 25 hours per week. These positions are normally compensated. Upon completion of the course, students will provide written documentation concerning their work projects.

Prerequisite: Upperclass standing in Computer Science with at least a 3.0 average in computer courses **or** permission of instructor.

Note: This course may not be taken concurrently with 63.495.



Mike Loomis, laptop support technician, works with students in the lobby of the Doyle Information Technology Center.