COMPUTER ENGINEERING AND COMPUTER SCIENCE

College of Engineering

Department Chair
Sandra Cynar

Department Office
Engineering and Computer Sciences (ECS) - 552

Telephone
(562) 985-4285

Website
http://www.cecs.csulb.edu/

Faculty

Professors
Joel Carissimo
Sandra Cynar
Wayne Dick
Edward Evans
Sheila Foster
Arthur Gittleman
Shui Lam
John Lane (Emeritus, 1999)
Dar-Biau Liu
Michael K. Mahoney
Carl Maltz
Tracy Bradley Maples
Thinh Nguyen
Morton Schwartz
Dennis Volper
Robert Wilson

Associate Professors
Todd Ebert
Tom Johnson
Alvaro Monge

Undergraduate Advisors
Joel Carissimo
Tom Johnson

Graduate Advisors
Wayne Dick
Dar-Biau Liu

Administrative Support Coordinator
Janet Leimer

View the CSU, Long Beach *Catalog* on-line at www.csulb.edu by clicking on "Academics" and then "CSULB Catalog."

Students desiring detailed information should contact the department for referral to one of the faculty advisors.

Accreditation

The Bachelor of Science in Computer Engineering is accredited by the EAC (Engineering Accrediting Commission) of Accreditation Board for Engineering and Technology, Inc. (ABET), 111 Market Place, Suite #1050, Baltimore, MD 21202, phone: 410/347-7710 (website: http://www.abet.org). The Bachelor of Science in Computer Science is accredited by the Computer Science Accreditation Commission (CSAC) of Computer Science Accreditation Board (CSAB),111 Market Place, Suite #1050, Baltimore, MD 21202, phone: 410/347-7703 (web site: http://csab.org).

Advisory Board

The Department of Computer Engineering and Computer Science is supported by an Advisory Board composed of engineers, computer scientists, and business executives in the Southern California area. This liaison helps the department keep its curricula responsive to the needs of the community.

Undergraduate Educational Objectives

All courses and programs in the Department of Computer Engineering and Computer Science are designed to meet the following objectives: Students will receive a comprehensive education in Computer Engineering and/or Computer Science, as well as the sciences and humanities, that will serve them in both their professional and personal lives. Students will be able to analyze and solve problems in Computer Engineering and/or Computer Science using modern techniques, skills, and tools. Students will be able to communicate effectively. Students will be able to design systems, components or processes that meet performance, cost, time, safety, and quality requirements. Students will understand their professional responsibilities, including remaining current in their field of study, and will be able to analyze the social and ethical implications of their work.

Undergraduate Degree Programs

The Department of Computer Engineering and Computer Science offers programs leading to a Bachelor of Science in Computer Engineering or a Bachelor of Science in Computer Science. The goal of both programs is to prepare graduates for a wide variety of computer-related careers by integrating the theoretical foundations of the discipline with practical applications. The degree in Computer Engineering focuses on computer hardware (design, construction, and operation of computer systems) while the Computer Science degree places more emphasis on computer software (programming). The high school student planning to enter either program is advised to pursue a strong program in science and mathematics.

Bachelor of Science in Computer Engineering (code 3-4013) (126 units)

The Bachelor of Science in Computer Engineering degree program has a two-part objective. The first is to provide students with a strong background in mathematics, physics, and engineering science. The curriculum includes a core of standard electrical engineering courses as well as courses in digital systems, computer organization, programming languages, and software, including operating systems, compiler theory and software engineering. The second objective is to provide students with the skills necessary to be effective contributors in a quality-oriented, customer-focused environment. Courses throughout the program, especially those in the junior and senior years, emphasize an open-ended, design-oriented approach to solving engineering problems. Teamwork, communication skills, and an interdisciplinary approach to problem solving are integrated into the senior, computer engineering design courses.

Requirements

Lower Division: CECS 174, 201, 228, 261, 274; MATH 122, 222; PHYS 151, 152 (or EE 210 and 210L).

Upper Division: CECS 301, 311, 325, 326, 340, 346, 347, 360, 440, 443, 460A, 460B; EE 380 (or MATH 380), 386; MATH 323, 370A; plus nine units of approved electives to be selected from CECS 406, 451, 474, 475, 497, EE 486, ENGR 310 (or ENGL 317).

Bachelor of Science in Computer Science (code 3-4011) (126 units)

The Bachelor of Science in Computer Science degree is designed to prepare graduates for a variety of professional careers in the computer field. The curriculum is designed to provide students with both breadth and depth in computer science. Breadth is achieved through a series of core courses that stress a balance between the theoretical and practical aspects of computer science. The topics covered in these courses include the following: the basics of programming languages, software design and analysis, data structures, algorithms, digital systems, computer organization, computer architecture, and operating systems. Extensive laboratory time is required for these courses, and design and analysis experiences are emphasized. Depth is achieved through courses (both required and elective) on advanced computer science topics. These courses provide students with in-depth knowledge of the material covered in the breadth portion of the curriculum.

Requirements

Lower Division: CECS 174, 201, 228, 261 or 281, 274; MATH 122, 222, 247; PHYS 151, 152 (or EE 210 and 210L); four units of approved courses in science or with strong emphasis in quantitative methods.

Upper Division: CECS 323, 325, 326, 328, 340, 424, 440, 443; ENGR 350; MATH 323, 380 (or EE 380); ENGR 310 or ENGL 317; plus nine units of approved electives to be selected as follows: three (3) units from CECS 471, 475, 481; and six (6) units from CECS 405, 406, 419, 423, 426, 428, 444, 448, 449, 451, 470, 472, 473, 476, 497.

Minor in Computer Science (code 0-4011)

The minor in Computer Science is designed to prepare students in other majors for careers in a wide variety of fields which require computer science expertise.

Requirements

A minimum of 21 units. CECS 174, 201, 228, 261 (or 281 or ENGR 202), 274, 323, and 325.

Master of Science in Computer Science

The Master of Science in Computer Science is offered by the Department of Computer Engineering and Computer Science, College of Engineering. Two options are offered:

1. Option in Computer Engineering
2. Option in Computer Science

The Option in Computer Engineering offers advanced study in the theory, analysis, design and applications of both computer hardware and software. The Option in Computer Science offers advanced study in modeling software systems, operating systems, compiler construction, and analysis of algorithms.

Admission Procedures

To be considered for admission the graduate applicant must have earned a bachelor's degree from an accredited institution, have been in good standing at the last institution attended, and have a grade point average (GPA) of at least 2.7 for the last 60 semester units (90 quarter units) attempted. The general Graduate Record Examination (GRE) is required. There is no separate department application, but applicants should submit a second set of transcripts to the department.

Prerequisites

Option in Computer Engineering (code 6-4010)

1. A bachelor's degree in computer science, engineering, or other appropriate discipline from an accredited college or university, with a minimum grade point average (GPA) of 2.7 in the last 60 semester units attempted.
2. Credit in the following courses or their equivalents: CECS 274, 301, 325, 326, 346, 347, 360, 440, 443, 460A, and MATH 370A.
3. Students must consult with the program graduate advisor prior to enrolling in any course for the program.

Option in Computer Science (code 6-4011)

1. A bachelor's degree in computer science, engineering, or other appropriate discipline from an accredited college or university, with a minimum grade point average (GPA) of 2.7 in the last 60 semester units attempted.
2. Credit in the following courses or their equivalents: CECS 228, 274, 323 (or 421), 325, 326, 328, 424, 440, 443, and MATH 380.
3. Students must consult with the program graduate advisor prior to enrolling in any course for the program.

Advancement to Candidacy

Students applying for advancement to candidacy must:

1. have completed all undergraduate deficiencies with grades of C or better;
2. have attained an overall grade point average (GPA) of 3.0;
3. have completed at least 12 units applicable to the degree with a GPA of at least 3.0;
4. have passed the University Writing Proficiency Examination;
5. and have their program of studies approved by the CECS department graduate advisor.

Requirements for the Option in Computer Engineering (code 6-4010)

Students must complete a minimum of 30 graduate and approved upper-division course units including the following:

1. At least 21 units at the graduate level of instruction;
2. CECS 526, 530, 531, and 546;
3. All students must complete either:

   A. a comprehensive examination, or

   B. a thesis with oral defense which requires a total of 6 units of CECS 697 or 698, of which at least 4 units must be CECS 698.

Requirements for the Option in Computer Science (code 6-4011)

Students must complete a minimum of 30 graduate and approved upper-division course units including the following:

1. At least 21 units at the graduate level of instruction;
2. CECS 526, 528, 530, and 543;
3. All students must complete either:
   A. a comprehensive examination, or
   B. a thesis with oral defense which requires a total of 6 units of CECS 697 or 698, of which at least 4 units must be CECS 698.

Courses (CECS)

Lower Division

110. Introduction to the Internet (3)
Prerequisite: Some computer experience. Provides a general overview of computer systems, networking, and the Internet. World-Wide Web, email, telnet, ftp, newsgroups, finding information on the Internet, and basic Web page creation. Considers legal, ethical, privacy and security issues on the Internet. (Lecture 2 hours, laboratory 3 hours.)

126. Introduction to the UNIX Operating System (3)
Prerequisite: Consent of instructor. UNIX operating system's user interface. File and directory commands, editor commands, communication commands and other UNIX utilities. UNIX access permissions and security, I/O redirection, pipes, metacharacters, and full-screen editing with vi. Electronic communication, simple shell programming, and using the Xwindow System. (Lecture 2 hours, laboratory 3 hours.)

174. Programming and Problem Solving I (3)
Prerequisite: MATH 117 (or equivalent.) Introduction to the basic concepts of computer science and the fundamental techniques for solving problems using the C++ programming language. Structured problem solving, object-oriented programming, programming style. Applications to numerical and non-numerical problems. Not open to students with credit in CECS 175. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

201. Computer Logic Design I (3)
Prerequisite: MATH 117 (or equivalent). Basic topics in combinational and sequential switching circuits with applications to the design of digital devices. Laboratory projects with Small Scale Integration (SSI) implementations using Computer Aided Design (CAD). (Lecture 2 hours, lab 3 hours.) Traditional grading only.

228. Discrete Structures with Computer Science Applications I (3)
Prerequisites: CECS 174 and MATH 122. The specification, development and analysis of algorithms. Sets, relations and functions. Logic and mathematical structures used in computer science. Introduction to combinatorics. Programming projects to exemplify these concepts. (Lecture 2 hours, laboratory/problem session 3 hours.) Traditional grading only.

261. Computing with Java (3)
Prerequisite: CECS 174. Comparison of basic Java constructs to similar constructs in C++. Object-oriented programming in Java. Applets and graphical user interfaces. Mouse, key, and window events. Exception handling and files. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

271. Introduction to Numerical Methods (3)
Prerequisites: CECS 174 and MATH 123. An introduction to numerical methods and the FORTRAN programming language. Analysis of computational errors; iterative methods for finding roots and for solving systems of equations. Numerical techniques for evaluating integrals, determining derivatives, and solving ordinary differential equations. FORTRAN programming projects will be assigned. Not open to students with credit in CECS 342. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

274. Programming and Problem Solving II (3)
Prerequisite: CECS 174. Disciplined methods of design, coding and testing using the C++ programming language. Data abstraction, object-oriented design. Introduction to data structures (linked lists, stacks, queues and trees.) Recursion. Sorting and searching. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

281. GUI Programming (3)
Prerequisite: CECS 174. Introduction to programming in a graphical user interface (GUI) environment. Menus, dialog boxes, forms, textboxes, toolbars and other controls. Properties, methods and events of objects. Program control statements including if, case, while, loop, and do. System objects including the mouse, screen and printer. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

Upper Division

301. Computer Logic Design II (3)
Prerequisite: CECS 174. Pre- or Corequisite: CECS 311 or EE 331. Sequential logic, programmable logic design, basic Arithmetic Logic Unit (ALU) design and memory devices. Laboratory projects with Medium Scale to Very Large Scale Integration (MSI to VLSI) implementations and Computer Aided Design (CAD). (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

311. Data Acquisition, Processing, and Display (3)
Prerequisites: CECS 201, EE 210 and 210L or PHYS 152. Hardware and software used to acquire, process, and display real-time signals. Transducers. Amplifiers. Buffers and isolators. Interrupt generators. Analog–to-digital and digital-analog converters. Display hardware. Software packages for electronics simulation and data acquisition. Sources of noise in digital systems. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

323. Database Fundamentals (3)
Prerequisites: CECS 228, 274, MATH 122, 222. Fundamental topics on database management. Topics include entity-relationship models, database design, data definition language, the relational model, data manipulation language, database application programming and normalization. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

325. Computer Organization and Assembly Language Programming (3)
Prerequisites: CECS 201 and 274. Basic computer organization, representation of information and instruction, addressing techniques, input/output, assembly language programming, macros and macro processing. Introduction to software systems, including assemblers, linkage editors and loaders. Programming assignments in assembly language. (Lecture 2 hours, lab 3 hours.) Traditional grading only.

326. Operating Systems (4)
Prerequisite: CECS 325. The structure and functions of operating systems. Interrupt handling, processes and interprocess communication, memory management, resource scheduling, information sharing and protection. Project implementation in C/C++. (Lecture 3 hours, laboratory 3 hours.) Traditional grading only.

328. Discrete Structures with Computer Science Applications II (3)
Prerequisites: CECS 228 and 274. A broad view of data structures and the structure-preserving operations on them. Abstract data types, algorithms, complexity. Programming projects to exemplify these concepts. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

340. Discrete Event Systems Modeling and Simulation (3)
Prerequisites: CECS 201 and 228. Modeling and simulation of discrete event systems specific to computer science and computer engineering including networks, queuing systems, digital logic, and computer architecture. The use of general purpose and specialized languages for these systems will be explored. Various methods for system input data (stochastic and deterministic) will be explored. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

346. Microprocessors and Applications (3)
Prerequisite: CECS 325. Study of microprocessor and microcomputer elements. Design of microprocessor–based systems to solve practical problems. Laboratory projects using CAD implementations and hardware design languages for simulation of designs. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

347. Microprocessor Hardware Design (3)
Prerequisites: CECS 301 and 346. Study of embedded processor applications and interfacing. Embedded systems design, control of external devices, embedded programming in C and assembly. Laboratory in implementation of embedded designs and hardware assisted debugging. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

360. Integrated Circuit Design Software (3)
Prerequisites: CECS 301, 325, MATH 123 or 222. Introduction to Computer Aided Design tools used in the design and fabrication of integrated circuits. Discussion of the IC fabrication process, the layout and routing of basic gates, transistor level design of gates, synthesis and RTL level design, floor planning, and IC development costs. (Lecture 2 hours, lab 3 hours.) Traditional grading only.

*405. Special Topics in Computer Science (3)
Prerequisite: Senior standing in computer science major. Selected topics from recent advances in computer science and technology. Course content will vary from year to year. (Lecture-problems 3 hours.) Repeatable to a maximum of 6 units with different topics. Traditional grading only.

*406. Special Topics in Computer Science (3)
Prerequisite: Senior standing in a computer science program. Each offering is based on an area in computer science and technology in which recent advances have been made. Specific topic will be recorded on student's transcript. (Lecture-problems 2 hours, laboratory 3 hours.) Repeatable to a maximum of 6 units with different topics. Traditional grading only.

*419. Theory of Computation (3)
Prerequisite: CECS 328. Finite automata and regular expressions. Pushdown automata and context-free languages. Turing machines and computability. Computational complexity. (Lecture-problems 3 hours.) Traditional grading only.

*423.. Database Applications (3)
Prerequisite: CECS 323 (or 421). Preparation for work on commercial database systems. Advanced modeling and analysis, data definition, constraints, retrieval, manipulation, security and usability. Commercial development tools, distributed/multi-tier environments and integration of databases with intranets and internets. Projects may include cooperation with students from other departments. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*424. Organization of Programming Languages (4)
Prerequisites: CECS 326 and 328. Comparison of programming languages (C, C++, Ada, Java, etc.) in their design and structure regarding features such as data types, control structures and runtime considerations. Includes computer projects illustrating some of these concepts. (Lecture 3 hours, laboratory 3 hours.) Traditional grading only.

*426. Topics in Operating Systems (3)
Prerequisite: CECS 326. Advanced operating system analysis and design. Topics of current interest. Project implementation (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*428. Analysis of Algorithms (3)
Prerequisite: CECS 328. Applications of standard combinatorial techniques to applied programming problems. Rigorous analysis of the correctness and complexity of algorithms. Advanced graph algorithms are emphasized. Topics include shortest paths on graphs, sorting, string matching, union find problem, divide-and-conquer technique, and weighted-edge problem. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

440. Computer Architecture (3)
Prerequisites: CECS 325 and 340. Review of logic design. Register transfer and micro-operations. Basic computer organization. Central processor organization. Microprogram control organization. Arithmetic processor design. Arithmetic algorithms. Input-output organization. Memory organization. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*443. Software Engineering (3)
Prerequisite: CECS 326. Software life cycle. Functional decomposition, data flow and object-oriented development. Reusability and portability. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*444. Compiler Construction (4)
Prerequisites: CECS 325 and 328. Syntax directed compiler study. Organization of a compiler and overall design: parsing, semantic analysis, optimization and code generation. (Lecture 3 hours, laboratory 3 hours.) Traditional grading only.

*446. Microprocessor Systems Design (3)
Corequisites: CECS 347 and 440. Computer system design using microprocessors including the use of multiple CPUs and RISC processors. (Lecture-problems 2 hours, laboratory 3 hours.) Traditional grading only.

*448. User Interface Design (3)
Prerequisites: CECS 323 (or 421), and any one of CECS 471, 475, 481. Evaluation, design and programming of user interface systems. Fundamentals of human cognition, system characteristics, and the interaction between humans and systems. Usability methods and user/task-centered design. Tools for designing and building user interfaces, with emphasis on rapid applications development. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*449. Computer Graphics (3)
Prerequisites: MATH 247, CECS 261 and 274. Basic software and hardware of 2-D computer graphics. Applications. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*451. Artificial Intelligence (3)
Prerequisites: CECS 228 and 274. Introduction to the principles and programming methods of artificial intelligence. Topics include symbol manipulation, knowledge representation, searching, expert systems and logic programming. Project implementation in LISP or Prolog. Not open to students with credit in CECS 420 or 450. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

460A. System on Chip Design I (3)
Prerequisite: CECS 360 or 440. A complete System on a Chip (SOC) design flow beginning with a design specification and ending with a working SOC. Creation of RTL level modules designed for reuse, integration of Intellectual Property (IP) for both RTL level and physical level IP, IC verification, and the creation of self-checking test benches for SOC designs. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

460B. System on Chip Design II (3)
Prerequisite: CECS 460A. System on a Chip (SOC) design applications. A variety of SOC designs will be built as class project which involve both the hardware and software aspect of a SOC along with hardware/software integration problems. Design reviews, design specification and team design implementation will be stressed along with project planning and tracking mechanisms for system level design problems. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*470. World–Wide Web Development (3)
Prerequisites: CECS 326 and 323 (or 421). Introduction to commercial practice in World-Wide Web site development. Web server and client software and the underlying networking technologies and protocols. Page and site design and programming using contemporary development tools and languages. Typography, graphics, database integration, usability, ethics and future directions. Projects will include site development for an actual client and may include cooperation with students from other departments. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*471. Internet Applications Using Java (3)
Prerequisites: CECS 261 and 274. Threads, images, and animation. User interface component libraries. Streams, readers and writers. Client-server applications. Java beans. Internationalization. Introduction to remote method invocation, Java database connectivity, and multimedia applications. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*472. Computer Network Programming (3)
Prerequisites: CECS 326 and 328. Fundamentals of client-server programming. Concepts of computer network programming including RPC and CORBA. Introduction to the principles of TCP and routing. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*473. Enterprise Web Applications (3)
Prerequisites: CECS 323 (or 421), 470, and any one of CECS 471, 475, 481. Web commerce, high-availability Web sites, and informa-

tion architecture. Advanced client side and server side scripting, advanced searching and indexing, application servers. Legal and business aspects. Projects may include cooperation with students from other departments. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*474. Computer Network Interoperability (3)
Prerequisite: CECS 326. An overview of computer network theory and practice from a systems perspective. Topics include network infrastructure, local area network (LAN) protocols, wide area network (WAN) protocols, switching technologies, Internet Protocol (IP), Transmission Control Protocol (TCP), network security, and network configuration, design, and performance. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*475. Object-Oriented Programming and C++ (3)
Prerequisite: CECS 274. An overview of object-oriented programming, data abstraction, and the C++ language. Classes, constructors, destructors, function and operator overloading, inheritance, polymorphism, input/output, standard template library (STL), templates, Visual C++, Microsoft Foundation Class (MFC), and object-oriented design. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*476. System and Network Administration (3)
Prerequisites: CECS 326 and 328. Introduction to the management and administration of Unix systems and TCP/IP networks. Managing users, local and network file systems, electronic mail, print queues. Establishing and managing a network. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

*481. Event Driven Programming (3)
Prerequisite: CECS 326. A rigorous introduction to Graphical User Interface (GUI) programming as provided in a number of popular software development products. Topics include Windows API (Application Programming Interface) using C and C++, Visual Basic and other development tools. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

496. Computer Science Problem Solving (1)
Prerequisites: CECS 274 and consent of instructor. A study of problem solving in Computer Science. Discussion of the theory necessary to solve computer science problems as well as the solutions to the problems. Problems studied will involve applications of graph theory, data structures, recursion, and algorithms. The aim of the course is to enhance the student's problem solving ability. (Lecture 1 hour.) Course may be repeated in different semesters to a maximum of 6 units. Traditional grading only.

*497. Directed Studies (1–3)
Prerequisite: Consent of instructor. Assigned study in topics in current computer literature or computer-related projects with a final report. May be repeated for a total of 6 units with written permission of the Department Chair.

Graduate Courses

521./621. Database Architecture (3)
Prerequisites: CECS 328 and 323 (or 421). (Master's students register in CECS 521 or 621; Ph.D. students register in CECS 621.) Relational database design theory-a rigorous approach. Security, recovery, transaction management, distributed databases and query optimization. Additional projects required for CECS 621. (Lecture-problems 3 hours.) Traditional grading only.

524./624. Advanced Topics in Programming Languages (3)
Prerequisite: CECS 424 and 471. (Master's students register in CECS 524 or 624; Ph.D. students register in CECS 624.) Intensive study of languages of current interest which support object-oriented, client-server, and multimedia applications (e.g. JAVA). Additional projects required for CECS 624. (Lecture-problems 3 hours.) Traditional grading only.

526./626. Advanced Operating Systems (3)
Prerequisites: CECS 228 and 326. (Master's students register in CECS 526 or 626; Ph.D. students register in CECS 626.) Theoretical foundations of concepts applied in the design of operating systems. Control of concurrent processes, deadlocks, mutual exclusion, virtual memory, resource management and scheduling. Additional projects required for CECS 626. (Lecture–problems 3 hours.) Traditional grading only.

528./628. Advanced Analysis of Algorithms (3)
Prerequisite: CECS 428. (Master's students register in CECS 528 or 628; Ph.D. students register in CECS 628.) Theoretical analysis of algorithms. Divide and conquer, dynamic programming and greedy algorithms; basic search and traversal techniques including search trees; sorting; matrix manipulations; NP–completeness. Additional projects required for CECS 628. (Lecture–problems 3 hours.) Traditional grading only.

529./629. Advanced Compiler Design (3)
Prerequisite: CECS 444. (Master's students register in CECS 529 or 629; Ph.D. students register in CECS 629.) Real–world and theoretical problems encountered by the compiler writer. Error handling, table management, the symbol table, run–time problems, code optimization, code generation and register allocation. Additional projects required for CECS 629. (Lecture–problems 3 hours.) Traditional grading only.

530./630. Advanced Computer Architecture I (3)
Prerequisite: CECS 440. (Master's students register in CECS 530 or 630; Ph.D. students register in CECS 630.) Fundamentals of computer architecture. Description of architecture and description languages. Basic computer design and central processor implementation. Memory hierarchy and input/output. Pipelining. Vector processor, multiprocessor systems and dataflow machines. Additional projects required for CECS 630. (Lecture–problems 3 hours.) Traditional grading only.

531./631. Advanced Computer Architecture II (3)
Prerequisite: CECS 530. (Master's students register in CECS 531 or 631; Ph.D. students register in CECS 631.) Advanced computer architecture with emphasis on parallel processing. Vector processors and multiprocessor systems. Dataflow computation. RISC/CISC. Hypercube. Parallel software. Applications in artificial intelligence, signal/image processing, neural network and optical computing. Additional projects required for CECS 631. (Lecture–problems 3 hours.) Traditional grading only.

543./643. Advanced Software Engineering (3)
Prerequisite: CECS 443. (Master's students register in CECS 543 or 643; Ph.D. students register in CECS 643.) Study of software engineering as a broad, problem–solving discipline. Includes structured programming and software project management. Additional projects required for CECS 643. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

544./644. Software Testing and Verification (3)
Prerequisite: CECS 543. Various types of software testing and verification techniques for software development including black box, white box, incremental, top-down and bottom-up, static and dynamic, performance, regression, thread, and stress testing. Discussion of object-oriented software testing with a hierarchical approach. Metrics in complexity for testing, test, and verification plan will be introduced. Automatic software testing and some case studies. Additional projects required for CECS 644. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

545./645. Software Architecture (3)
Prerequisite: CECS 543. Topics include architectural styles, pipes and filters, data abstraction and object-oriented organization, event-based, implicit invocation, layered systems, repositories, interpreters, process control, distributed processes, domain-specific software architectures, and heterogeneous architecture. Component-based design patterns and some case studies. Additional projects required for CECS 645. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

546./646. Fault Tolerant Computing Systems (3)
Prerequisite: CECS 530. (Master's students register in CECS 546 or 646; Ph.D. students register in CECS 646.) Fault tolerant techniques are studied as tools to assure the reliability and continuous availability of computing systems. Case studies of modern fault tolerant systems reviewed. Software fault tolerant systems studied as alternatives to verification and validation approaches to software reliability. Additional projects required for CECS 646. (Lecture–problems 3 hours.) Traditional grading only.

549./649. Advanced Computer Graphics (3)
Prerequisite: CECS 449. (Master's students register in CECS 549 or 649; Ph.D. students register in CECS 649.) Three-dimensional representations, transformations and viewing. Color models and modeling methods. Hidden-line and hidden-surface removal. Lighting and shading. Visual realism. Topics of current interest. Additional projects required for CECS 649. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

551./651. Advanced Artificial Intelligence (3)
Prerequisite: CECS 451. (Master's students register in CECS 551 or 651; Ph.D. students register in CECS 651.) Advanced concepts in artificial intelligence. Topics include knowledge acquisition and representation, fuzzy logic, logical reasoning, multi-sensor integration, Dempster-Shafer's theory of evidential reasoning, real-time expert systems and neural networks. Additional projects required for CECS 651. (Lecture-problems 3 hours.) Traditional grading only.

552./652. Computer Simulation and Modeling (3)
Prerequisites: EE 380 (or MATH 380) and CECS 326. (Master's students register in CECS 552 or 652; Ph.D. students register in CECS 652.) Studies of general purpose and special simulation software. Model verification including graphical models Applications in various areas. Additional projects required for CECS 652. (Lecture-problems 3 hours.) Traditional grading only.

553./653. Machine Vision (3)
Prerequisite: Graduate standing in engineering or computer science. (Master's students register in CECS 553 or 653; Ph.D. students register in CECS 653.) Discussion and laboratory implementation of current research in vision and image understanding. Topics include image formation, early processing, segmentation, relational structures in 2-D and 3-D, motion, stereo, 3-D reconstruction, morphological methods and computer architecture for machine vision. Additional projects required for CECS 653. (Lecture-problems 2 hours, laboratory 3 hours.) Traditional grading only.

570./670. Concurrent Parallel Programming (3)
Prerequisite: CECS 428 or 440. (Master's students register in CECS 570 or 670; Ph.D. students register in CECS 670.) An introduction to concurrent and parallel programming for multiprocessing as well as distributed systems. Computational models and paradigms. Parallel programming languages and programming tools. Portable parallel programming and mapping techniques. Heterogeneous concurrent programming. Concurrent programming on local networks on workstations and personal computers. Additional projects required for CECS 670. (Lecture-problems 2 hours, laboratory 3 hours.) Traditional grading only.

572./672. Network Architecture (3)
Prerequisite: CECS 472 or 474. (Master's students register in CECS 572 or 672; Ph.D. students register in CECS 672.) Advanced concepts in computer networking. Distributed architectures, computer network standards and design, and computer network performance issues. Additional projects required for CECS 672. (Lecture-problems 3 hours.) Traditional grading only.

575./675. Object–Oriented Analysis and Design (3)
Prerequisite: CECS 475 and either CECS 443 or 543. (Master's students register in CECS 575 or 675; Ph.D. students register in CECS 675.) An object–oriented approach to software development based on modeling objects from the real world and then using the model to build a language–independent design organized around those objects. Object–oriented methodology from problem statement through analysis, system design, and object design. Implementation of object–oriented designs in various target environments. Case studies. Additional projects required for CECS 675. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

590./690. Special Topics in Computer Science (3)
Prerequisites: Graduate standing and consent of instructor. (Master's students register in CECS 590 or 690; Ph.D. students register in CECS 690.) Each offering is based on an area in computer science and technology in which recent advances have been made. Additional projects required for CECS 690. (Lecture-problems 3 hours.) Repeatable to a maximum of 6 units with consent of department. Traditional grading only.

621./521. Database Architecture (3)
Prerequisites: CECS 328 and 323 (or 421). (Master's students register in CECS 521 or 621; Ph.D. students register in CECS 621.) Relational database design theory-a rigorous approach. Security, recovery, transaction management, distributed databases and query optimization. Additional projects required for CECS 621. (Lecture-problems 3 hours.) Traditional grading only.

624./524. Advanced Topics in Programming Languages (3)
Prerequisite: CECS 424 and 471. (Master's students register in CECS 524 or 624; Ph.D. students register in CECS 624.) Intensive study of languages of current interest which support object-oriented, client-server, and multimedia applications (e.g. JAVA). Additional projects required for CECS 624. (Lecture-problems 3 hours.) Traditional grading only.

626./526. Advanced Operating Systems (3)
Prerequisites: CECS 228 and 326. (Master's students register in CECS 526 or 626; Ph.D. students register in CECS 626.) Theoretical foundations of concepts applied in the design of operating systems. Control of concurrent processes, deadlocks, mutual exclusion, virtual memory, resource management and scheduling. Additional projects required for CECS 626. (Lecture–problems 3 hours.) Traditional grading only.

628./528. Advanced Analysis of Algorithms (3)
Prerequisite: CECS 428. (Master's students register in CECS 528 or 628; Ph.D. students register in CECS 628.) Theoretical analysis of algorithms. Divide and conquer, dynamic programming and greedy algorithms; basic search and traversal techniques including search trees; sorting; matrix manipulations; NP–completeness. Additional projects required for CECS 628. (Lecture–problems 3 hours.) Traditional grading only.

629./529. Advanced Compiler Design (3)
Prerequisite: CECS 444. (Master's students register in CECS 529 or 629; Ph.D. students register in CECS 629.) Real–world and theoretical problems encountered by the compiler writer. Error handling, table management, the symbol table, run–time problems, code optimization, code generation and register allocation. Additional projects required for CECS 629. (Lecture–problems 3 hours.) Traditional grading only.

630./530. Advanced Computer Architecture I (3)
Prerequisite: CECS 440. (Master's students register in CECS 530 or 630; Ph.D. students register in CECS 630.) Fundamentals of computer architecture. Description of architecture and description languages. Basic computer design and central processor implementation. Memory hierarchy and input/output. Pipelining. Vector processor, multiprocessor systems and dataflow machines. Additional projects required for CECS 630. (Lecture–problems 3 hours.) Traditional grading only.

631./531. Advanced Computer Architecture II (3)
Prerequisite: CECS 530. (Master's students register in CECS 531 or 631; Ph.D. students register in CECS 631.) Advanced computer architecture with emphasis on parallel processing. Vector processors and multiprocessor systems. Dataflow computation. RISC/CISC. Hypercube. Parallel software. Applications in artificial intelligence, signal/image processing, neural network and optical computing. Additional projects required for CECS 631. (Lecture–problems 3 hours.) Traditional grading only.

643./543. Advanced Software Engineering (3)
Prerequisite: CECS 443. (Master's students register in CECS 543 or 643; Ph.D. students register in CECS 643.) Study of software engineering as a broad, problem–solving discipline. Includes structured programming and software project management. Additional projects required for CECS 643. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

644./544. Software Testing and Verification (3)
Prerequisite: CECS 543. Various types of software testing and verification techniques for software development including black box, white box, incremental, top-down and bottom-up, static and dynamic, performance, regression, thread, and stress testing. Discussion of object-oriented software testing with a hierarchical approach. Metrics in complexity for testing, test, and verification plan will be introduced. Automatic software testing and some

case studies. Additional projects required for CECS 644. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

645./545. Software Architecture (3)
Prerequisite: CECS 543. Topics include architectural styles, pipes and filters, data abstraction and object-oriented organization, event-based, implicit invocation, layered systems, repositories, interpreters, process control, distributed processes, domain-specific software architectures, and heterogeneous architecture. Component-based design patterns and some case studies. Additional projects required for CECS 645. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

646./546. Fault Tolerant Computing Systems (3)
Prerequisite: CECS 530. (Master's students register in CECS 546 or 646; Ph.D. students register in CECS 646.) Fault tolerant techniques are studied as tools to assure the reliability and continuous availability of computing systems. Case studies of modern fault tolerant systems reviewed. Software fault tolerant systems studied as alternatives to verification and validation approaches to software reliability. Additional projects required for CECS 646. (Lecture–problems 3 hours.) Traditional grading only.

649./549. Advanced Computer Graphics (3)
Prerequisite: CECS 449. (Master's students register in CECS 549 or 649; Ph.D. students register in CECS 649.) Three-dimensional representations, transformations and viewing. Color models and modeling methods. Hidden-line and hidden-surface removal. Lighting and shading. Visual realism. Topics of current interest. Additional projects required for CECS 649. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

650./750. Pattern Recognition Using Artificial Intelligence (3)
(Master's students register in CECS 650; Ph.D. students register in CECS 750.) General concepts of pattern recognition and trainable classifiers, decision theory, supervised learning, non-parametric techniques, rule-based systems and neural networks. Additional projects required for Ph.D. students. (Lecture-problems 2 hours, laboratory 3 hours.) Traditional grading only.

651./551. Advanced Artificial Intelligence (3)
Prerequisite: CECS 451. (Master's students register in CECS 551 or 651; Ph.D. students register in CECS 651.) Advanced concepts in artificial intelligence. Topics include knowledge acquisition and representation, fuzzy logic, logical reasoning, multi-sensor integration, Dempster-Shafer's theory of evidential reasoning, real-time expert systems and neural networks. Additional projects required for CECS 651. (Lecture-problems 3 hours.) Traditional grading only.

652./552. Computer Simulation and Modeling (3)
Prerequisites: EE 380 (or MATH 380) and CECS 326. (Master's students register in CECS 552 or 652; Ph.D. students register in CECS 652.) Studies of general purpose and special simulation software. Model verification including graphical models Applications in various areas. Additional projects required for CECS 652. (Lecture-problems 3 hours.) Traditional grading only.

653./553. Machine Vision (3)
Prerequisite: Graduate standing in engineering or computer science. (Master's students register in CECS 553 or 653; Ph.D. students register in CECS 653.) Discussion and laboratory implementation of current research in vision and image understanding. Topics include image formation, early processing, segmentation, relational structures in 2-D and 3-D, motion, stereo, 3-D reconstruction, morphological methods and computer architecture for machine vision. Additional projects required for CECS 653. (Lecture-problems 2 hours, laboratory 3 hours.) Traditional grading only.

670./570. Concurrent Parallel Programming (3)
Prerequisite: CECS 428 or 440. (Master's students register in CECS 570 or 670; Ph.D. students register in CECS 670.) An introduction to concurrent and parallel programming for multiprocessing as well as distributed systems. Computational models and paradigms. Parallel programming languages and programming tools. Portable parallel programming and mapping techniques. Heterogeneous concurrent programming. Concurrent programming on local networks on workstations and personal computers. Additional projects required for CECS 670. (Lecture-problems 2 hours, laboratory 3 hours.) Traditional grading only.

672./572. Network Architecture (3)
Prerequisite: CECS 472 or 474. (Master's students register in CECS 572 or 672; Ph.D. students register in CECS 672.) Advanced concepts in computer networking. Distributed architectures, computer network standards and design, and computer network performance issues. Additional projects required for CECS 672. (Lecture-problems 3 hours.) Traditional grading only.

673./773. Topics in Distributed Computer Systems (3)
Prerequisite: CECS 572 or 672. (Master's students register in CECS 673; Ph.D. students register in CECS 773.) Network operating systems vs distributed operating systems, research and design issues of distributed operating systems, resources and resource management in distributed systems, communication security and user authentication. Additional projects required for Ph.D. students. (Lecture–problems 3 hours.) Traditional grading only.

675./575. Object–Oriented Analysis and Design (3)
Prerequisite: CECS 475 and either CECS 443 or 543. (Master's students register in CECS 575 or 675; Ph.D. students register in CECS 675.) An object–oriented approach to software development based on modeling objects from the real world and then using the model to build a language–independent design organized around those objects. Object–oriented methodology from problem statement through analysis, system design, and object design. Implementation of object–oriented designs in various target environments. Case studies. Additional projects required for CECS 675. (Lecture 2 hours, laboratory 3 hours.) Traditional grading only.

690./590. Special Topics in Computer Science (3)
Prerequisites: Graduate standing and consent of instructor. (Master's students register in CECS 590 or 690; Ph.D. students register in CECS 690.) Each offering is based on an area in computer science and technology in which recent advances have been made. Additional projects required for CECS 690. (Lecture-problems 3 hours.) Repeatable to a maximum of 6 units with consent of department. Traditional grading only.

694. Seminar in Computer Science (3)
Prerequisite: Six units of 500 or 600 level CECS courses. Intensive study of a broad selection of conceptual and theoretical problems in computer science. A written student research project and an oral presentation are required. Traditional grading only.

697. Directed Research (1-3)
Prerequisite: Classified Graduate standing. Theoretical and experimental problems in computer science and engineering requiring intensive analysis. (Independent Study.) Traditional grading only.

698. Thesis or Industrial Project (2-6)
Prerequisite: Advancement to Candidacy. Planning, preparation, and completion of a thesis or equivalent industrial project report on a suitable topic in computer engineering and computer science following the library's prescribed format. Requires consultation with department's Graduate Advisor and submission of an Agreement for Independent Study form each semester of enrollment. May be repeated to a total of 6 units.

750./650. Pattern Recognition using Artificial Intelligence (3)
(Master's students register in CECS 650; Ph.D. students register in CECS 750.) General concepts of pattern recognition and trainable classifiers, decision theory, supervised learning, non-parametric techniques, rule-based systems and neural networks. Additional projects required for Ph.D. students. (Lecture-problems 2 hours, laboratory 3 hours.) Traditional grading only.

773./673. Topics in Distributed Computer Systems (3)
Prerequisite: CECS 572 or 672. (Master's students register in CECS 673; Ph.D. students register in CECS 773.) Network operating systems vs distributed operating systems, research and design issues of distributed operating systems, resources and resource management in distributed systems, communication security and user authentication. Additional projects required for Ph.D. students. (Lecture–problems 3 hours.) Traditional grading only.