CS   Computer Science

CS 100 THE COMPUTER SCIENCE PROFESSION. (1)

An introductory seminar which covers the fundamental activities, principles, and ethics of the computer science profession. An overview of the discipline of computer science, examples of careers, the history of computing and experience with elementary computing tools are included.

CS 101 INTRODUCTION TO COMPUTING I. (3)

An introduction to computing and its impact on society from a user's perspective. Topics include computation using spreadsheets, beautification using text formatters and word processors, information management with database managers, and problem solving through program design and implementation using a simple programming language. Not open to students who have received credit for higher level computer science courses.

CS 115 INTRODUCTION TO COMPUTER PROGRAMMING. (3)

This course teaches introductory skills in computer programming using a high-level computer programming language. There is an emphasis on both the principles and practice of computer programming. Covers principles of problem solving by computer and requires completion of a number of programming assignments.

CS 215 INTRODUCTION TO PROGRAM DESIGN, ABSTRACTION, AND PROBLEM SOLVING. (4)

The course covers introductory object-oriented problem solving, design, and programming engineering. Fundamental elements of data structures and algorithm design will be addressed. An equally balanced effort will be devoted to the three main threads in the course: concepts, programming language skills, and rudiments of object-oriented programming and software engineering. Prereq: EGR 102 or CS 115 or equivalent.

CS 216 INTRODUCTION TO SOFTWARE ENGINEERING TECHNIQUES. (3)

Implementation of large programming projects using object-oriented design techniques and software tools in a modern development environment. Software engineering topics to include: life cycles, metrics, requirements specifications, design methodologies, validation and verification, testing, reliability and project planning. Prereq: CS 215.

CS 221 FIRST COURSE IN COMPUTER SCIENCE FOR ENGINEERS. (2)

Characteristics of a procedure-oriented language; description of a computer as to internal structure and the representation of information; introduction to algorithms. Emphasis will be placed on the solution of characteristic problems arising in engineering. Prereq: Not open to students who have received credit for EGR 102 or CS 115.

CS 261 SOCIAL NETWORKS: METHODS AND TOOLS. (3)

The complex connectedness of the modern society is a multifaceted phenomenon resulting from the growing density of the human population, the advent of fast global mass transportation infrastructure, the emergence of global companies and markets, and spurred by the Internet and its applications such as the Web, Facebook and Twitter. In this course, we learn about graph theory, game theory and computational tools required to model and analyze social networks, matching markets, web search, network externalities, tipping points, information cascades, epidemics, small worlds, and voting schemes. The course requires no programming background and has no university-level prerequisites.

*CS 270 SYSTEMS PROGRAMMING. (3)

This course provides an introduction to computer systems and explores computer architecture, operating systems, and networks from a programmer's perspective. The course also introduces advanced programming and debugging tools. Topics include hardware instruction sets, machine language and C language program representations, linking/loading, operating systems (process management, scheduling, memory management, interprocess communication, and file systems), network programming (socket programming and web protocols), and common security attacks and solutions. Prereq: CS 216; coreq: EE 280.

CS 275 DISCRETE MATHEMATICS. (4)

Topics in discrete math aimed at applications in Computer Science. Fundamental principles: set theory, induction, relations, functions, Boolean algebra. Techniques of counting: permutations, combinations, recurrences, algorithms to generate them. Introduction to graphs and trees. Prereq: MA 113 and either EGR 102 or CS 115.

KEY: # = new course   * = course changed   † = course dropped

CS   Computer Science

CS 315 ALGORITHM DESIGN AND ANALYSIS.   (3)
Introduction to the design and analysis of algorithms. Asymptotic analysis of time complexity. Proofs of correctness. Algorithms and advanced data structures for searching and sorting lists, graph algorithms, numeric algorithms, and string algorithms. Polynomial time computation and NP-completeness. Prereq: CS 215, CS 275, and engineering standing.

CS 316 WEB PROGRAMMING.   (3)
This course introduces students to the World Wide Web, languages and techniques used for web programming, data transfer over the Internet, and the tools available in the web environment. Prereq: CS 216.

CS 321 INTRODUCTION TO NUMERICAL METHODS.   (3)
Floating point arithmetic. Numerical linear algebra: elimination with partial pivoting and scaling. Polynomial and piecewise interpolation. Least squares approximation. Numerical integration. Roots of nonlinear equations. Ordinary differential equations. Laboratory exercises using software packages available at computer center. Prereq: MA 114 and knowledge of a procedural computer language is required. (Same as MA 321.)

CS 335 GRAPHICS AND MULTIMEDIA.   (3)
This course focuses on the graphical human-machine interface, covering the principles of windowing systems, graphical interface design and implementation, and processing graphical data. There is an emphasis on medium-scale programming projects with graphical user interfaces using a high-level procedural programming language and concepts such as object-oriented design. Prereq: CS 216 and engineering standing.

CS 340 APPLICABLE ALGEBRA.   (3)
Topics include: Euclid's algorithm, unique factorization moduli arithmetic, Fermat's and Euler's theorems, Chinese remainder theorem, RSA public key encryption, Pollard rho factoring, pseudo primes, error correcting codes, Hamming codes, polynomial rings and quotient rings, field extensions, finite fields and BCH codes. Prereq: MA 322 or MA 213. (Same as MA 340.)

CS 371 INTRODUCTION TO COMPUTER NETWORKING.   (3)
Introduction to the principles and concepts of the Internet; data communications and digital channel characteristics; networking applications and protocols, client-server paradigm and network programming; reliable data transfer, end-to-end transport; addressing, forwarding and routing, datagram networks; media access control, data link control; selected topics from cloud computing, network security and network management. Concepts are combined with programming and other hands-on assignments to enhance the learning of these topics. Prereq: CS 270, CS 315.

CS 375 LOGIC AND THEORY OF COMPUTING.   (3)
Topics in logic and discrete math aimed at applications in Computer Science. Propositional calculus: truth tables, logical relations, proofs, tautologies, soundness. Predicate calculus: variables, quantifiers, equivalencies. Models of computation: logic circuits, finite automata, Turing machines. Prereq: MA 113, CS 215, CS 275 and engineering standing.

CS 378 INTRODUCTION TO CRYPTOLOGY.   (3)
The study of secrecy in digital systems. Methods of keeping information secure from classical systems dating from ancient times to modern systems based on modern mathematics. Basic methods of encryption using public key systems, block ciphers, and stream ciphers. The mathematical tools for the design and analysis of such systems. Topics will include classical cryptography, modern methods of public and private key encryption, authentication and digital signatures, hashing, and passwords. Number theory, abstract algebra, combinatorics, and complexity theory necessary for the design and analysis of cryptographic systems. Prereq: CS 315 and STA 281, or instructor's consent.

*CS 380 COMPUTER ORGANIZATION.   (3)
Hardware and software organization of a typical computer; machine language and assembler language programming, interfacing peripheral devices, and input-output programming; real-time computer applications, laboratory included. Prereq: Engineering standing, CS 215 and EE/CPE 282 or EE 280. (Same as CPE/EE 380.)

CS 383 INTRODUCTION TO EMBEDDED SYSTEMS.   (3)
A course in the hardware and software of microprocessors. Assembly language programming, address decoding, hardware interrupts, parallel and serial interfacing with various special purpose integrated circuits. Each student is expected to do homework assignments using microprocessor hardware. Prereq: EE 280 and EE/CS 380. (Same as EE 383.)

KEY: # = new course   * = course changed   † = course dropped

CS   Computer Science

*CS 395 INDEPENDENT WORK IN COMPUTER SCIENCE.   (1-3)

A course that provides an opportunity for supervised individual research and study in computer science. A topic of the course must be approved by a supervising instructor and the Computer Science Director of Undergraduate Studies. May be repeated to a maximum of six credits. Prereq: Consent of the Computer Science Department.

CS 405G INTRODUCTION TO DATABASE SYSTEMS.   (3)

Study of fundamental concepts behind the design, implementation and application of database systems. Brief review of entity-relationship, hierarchical and network database models and an in-depth coverage of the relational model including relational algebra and calculi, relational database theory, concepts in schema design and commercial database languages. Prereq: CS 315 and graduate or engineering standing.

CS 415G COMBINATORICS AND GRAPH THEORY.   (3)

A basic course in the theory of counting and graph theory. Topics in enumerative combinatorics may include: generating functions, compositions, partitions, Fibonacci numbers, permutations, cycle structure of permutations, permutations statistics, Stirling numbers of the first and second kind, Bell numbers, inclusion-exclusion. Topics in graph theory may include: Eulerian and Hamiltonian cycles, matrix tree theorem, planar graphs and the 4-color theorem, chromatic polynomial, Hall's marriage theorem, stable marriage theorem, Ramsey theory, electrical networks. Prereq: MA 213 or MA 322. (Same as MA 415G.)

*CS 416G INTRODUCTION TO OPTIMIZATION.   (3)

The course is an introduction to modern operations research and includes discussion of modeling, linear programming, dynamic programming, integer programming, scheduling and inventory problems and network algorithms. Prereq: MA 213 and MA 322 (or equivalent classes), or graduate student status, or consent of the department. (Same as MA 416G.)

CS 422 NUMERICAL SOLUTIONS OF EQUATIONS.   (3)

Linear equations: Gaussian elimination, special linear systems, orthogonalization, eigenproblem, iterative methods. Nonlinear equations: solutions of equations in one variable, solutions of systems of nonlinear equations. Optimization. Prereq: CS/MA 321 and MA 322; or consent of instructor. (Same as MA 422.)

CS 441G COMPILERS FOR ALGORITHMIC LANGUAGES.   (3)

The techniques of processing, specifying, and translating high-level computer languages are studied. Topics include finite state machines and lexical analysis, context-free grammars for language specification, attributed translation grammars, language parsing, and automatic generation of compilers by SLR, LALR, and other methods of analyzing context-free grammars. Other topics may include code optimization, semantics of programming languages, and top-down parsing. Prereq: CS 315 and engineering standing.

CS 450G FUNDAMENTALS OF PROGRAMMING LANGUAGES.   (3)

An intensive study of fundamental programming concepts exhibited in current high level languages. Concepts include recursion, iteration, coroutines, multiprocessing, backtracking, pattern-matching, parameter passing methods, data structures, and storage management. Typical languages studied are SNOBOL, LISP, PASCAL, and APL. Prereq: CS 370. Restricted to computer science and electrical engineering majors. Others by permission.

CS 460G MACHINE LEARNING.   (3)

Study of computational principles and techniques that enable software systems to improve their performance by learning from data. Focus on fundamental algorithms, mathematical models and programming techniques used in Machine Learning. Topics include: different learning settings (such as supervised, unsupervised and reinforcement learning), various learning algorithms (such as decision trees, neural networks, k-NN, boosting, SVM, k-means) and crosscutting issues of generalization, data representation, feature selection, model fitting and optimization. The course covers both theory and practice, including programming and written assignments that utilize concepts covered in lectures. Prereq: Strong programming ability (CS 315), basic probability and statistics (STA 281), and basic concepts of linear algebra (MA/CS 321 or MA/CS 322), or instructor's consent.

CS 463G INTRODUCTION TO ARTIFICIAL INTELLIGENCE.   (3)

The course covers basic techniques of artificial intelligence. The topics in this course are: search and game-playing, logic systems and automated reasoning, knowledge representation, intelligent agents, planning, reasoning under uncertainty, and declarative programming languages. The course covers both theory and practice, including programming assignments that utilize concepts covered in lectures. Prereq: CS 315, CS 375, and engineering standing.

KEY: # = new course   * = course changed   † = course dropped

CS   Computer Science

CS 470G INTRODUCTION TO OPERATING SYSTEMS.   (3)

This course provides an introduction and overview of operating system design, internals, and administration. Topics include classical operating systems (process management, scheduling, memory management, device drivers, file systems), modern operating systems concepts (kernel/microkernel designs, concurrency, synchronization, interprocess communication, security and protection), and operating system administration. Prereq: CS 315, CS 380, and graduate or engineering standing.

CS 471G NETWORKING AND DISTRIBUTED OPERATING SYSTEMS.   (3)

Broad overview of concepts in networking and distributed operating systems with examples. Topics will include protocol stacks, link, network, transport, and application layers, network management, the client-server model, remote procedure calls, and case studies of distributed OS and file systems. Prereq: CS 315 and graduate or engineering standing.

CS 480G ADVANCED COMPUTER ARCHITECTURE.   (3)

This course focuses on advanced computer architectures and low-level system software. Topics include RISC architectures, vector and multiprocessor architectures, multiprocessor memory architectures, and multiprocessor interconnection networks. Peripheral devices such as disk arrays, NICs, and video/audio devices are covered. Topics also include device drivers, interrupt processing, advanced assembly language programming techniques, assemblers, linkers, and loaders. Prereq: CPE/CS/EE 380. (Same as CPE/EE 480.)

CS 485G TOPICS IN COMPUTER SCIENCE (Subtitle required).   (2-4)

Studies of emerging research and methods in computer science. A review and extension of selected topics in the current literature. When the course is offered, a specific title with specific credits, the number of hours in lecture-discussion and laboratory will be announced. Lecture/discussion, two-four hours; laboratory, zero-four hours per week. May be repeated up to the discretion of the department.

CS 498 SOFTWARE ENGINEERING FOR SENIOR PROJECT.   (3)

Current approaches – practice and technologies – for developing reliable software: specifications, testing, and verification. Individual and team assignments focused on applying these approaches to software systems. A significant communication and composition component related to specifying, designing, presenting, and documenting software systems. Prereq: CS 115, 215, 216.

CS 499 SENIOR DESIGN PROJECT.   (3)

Projects to design and implement complex systems of current interest to computer scientists. Students will work in small groups. Prereq: CS 315 and engineering standing. This course is a Graduation Composition and Communication Requirement (GCCR) course in certain programs, and hence is not likely to be eligible for automatic transfer credit to UK.

CS 505 INTERMEDIATE TOPICS IN DATABASE SYSTEMS.   (3)

The course introduces a variety of modern techniques in database and distributed database systems. The major topics include, but are not limited to: object-oriented database systems; distributed, heterogeneous and web-based databases; knowledge based systems; physical database design; and security. The course covers a variety of methods that allow for a solution of database problems where the traditional relational database techniques are not viable or not sufficient. Prereq: CS 405 or consent of instructor.

CS 515 ALGORITHM DESIGN.   (3)

The design and analysis of efficient algorithms on data structures for problems in sorting, searching, graph theory, combinatorial optimization, computational geometry, and algebraic computation. Algorithm design techniques: divide-and-conquer, dynamic programming, greedy method, and randomization, approximation algorithms. Prereq: CS 315 and engineering standing.

CS 521 COMPUTATIONAL SCIENCES.   (3)

Study of computer science techniques and tools that support computational sciences and engineering. Emphasis on visualization, performance evaluation, parallel computing, and distributed computing. Prereq: Either EGR 102 or CS 115, and CS 380/EE 380 and engineering standing.

CS 522 MATRIX THEORY AND NUMERICAL LINEAR ALGEBRA I.   (3)

Review of basic linear algebra from a constructive and geometric point of view. Factorizations of Gauss, Cholesky and Gram-Schmidt. Determinants. Linear least squares problems. Rounding error analysis. Stable methods for updating matrix factorizations and for linear programming. Introduction to Hermitian eigenvalue problems and the singular value decomposition via the QR algorithm and the Lanczos process. Method of conjugate gradients. Prereq: MA 322. (Same as MA 522.)

KEY: # = new course   * = course changed   † = course dropped

CS   Computer Science

CS 535 INTERMEDIATE COMPUTER GRAPHICS.   (3)
Three-dimensional graphics primitives such as 3D viewing, lighting, shading, hidden line/surface removal, and more advanced topics such as solid modeling, image storage and representation, advanced raster graphics architecture and algorithms, advanced modeling techniques, and animation will be covered. Prereq: CS 335, CS 315, CS 321, and engineering standing.

CS 536 SITUATED COMPUTING.   (3)
This course covers the fundamental concepts involved in understanding and engineering a closed-loop, sensing, reasoning, and actuating agent. Biological models of sensing and actuation will be discussed and related to modern artificial counterparts. The course consists of three major topic areas: vision, brain, and robotics. It will introduce students to the issues in computer and biological vision, to models of belief representation and modification, architectures for percept processing and reasoning, machine learning for vision, neural networks, path planning, intelligent localization based on visual cues, and to forward and inverse kinematics, intelligent grasping, and the integration of perception and action. Prereq: CS 460G or consent of instructor.

CS 537 NUMERICAL ANALYSIS.   (3)
Floating point arithmetic. Direct methods for the solution of systems of linear algebraic equations. Polynomial and piecewise polynomial approximation, orthogonal polynomials. Numerical integration: Newton Cotes formulas and Gaussian quadrature. Basic methods for initial value problems for ordinary differential equations. The emphasis throughout is on the understanding and use of software packages for the solution of commonly occurring problems in science and engineering. Prereq: CS/MA 321 or equivalent or graduate standing or consent of instructor. Knowledge of a procedural computer language is required. (Same as EGR/MA 537.)

CS 541 COMPILER DESIGN.   (3)
Intermediate aspects of a compilation process with an emphasis on front-end issues. Practical issues in using compiler writing tools. Code generation for expressions, control statements and procedures (including parameter passing). Symbol tables, runtime organization for simple and structured variables. Using compilers and translators for automation (filters, programs writing programs). Prereq: CS 441 or consent of instructor.

CS 555 DECLARATIVE PROGRAMMING.   (3)
The course covers fundamentals of propositional and predicate logic, and their uses in declarative programming to model and solve computational problems. Topics include propositional satisfiability, satisfiability testing techniques such as the DPLL algorithm, automated reasoning techniques for predicate logic such as resolution with unification and logic programming. Prereq: CS 315 and CS 375 or consent of instructor.

#CS 564 COMPUTER SECURITY.   (3)
This course will introduce students to the basics of computer and software security. It will expose students to topics such as cryptography, secure hash functions, access control models, audit of computer systems, attacks on computer systems and countermeasures, elements of computer forensics, and elements of database and network security. Prereq: CS 270 or EE 287 or consent of the instructor.

CS 570 MODERN OPERATING SYSTEMS.   (3)
Brief review of classical operating system concepts (process and memory management, process coordination, device drivers, file systems, starvation/deadlock). Modern topics of files systems (log-structured file systems, distributed file systems, memory-based file systems), operating system design (monolithic, communication-kernel, extensible/adaptable, distributed shared memory), multiprocessor issues (scheduling, synchronization, IPC), security (internet attacks, encryption, defenses). Inspection and modification of actual operating system code (Linux). Prereq: CS 470 and engineering standing.

CS 571 COMPUTER NETWORKS.   (3)
Principles of computer networks using current Internet technologies and protocols as examples. Routing algorithms and protocols; end-to-end transport; flow control; congestion avoidance and control; mail, web, and file transfer protocols; designing and implementing applications using common network APIs. Advanced topics, included as time permits, include network security, multicast, and quality of service. Prereq: CS 471G or consent of instructor.

KEY: # = new course   * = course changed   † = course dropped

CS   Computer Science

#CS 572 NETWORK SECURITY. (3)

This course introduces students to the state of the art of network security problems and solutions. Topics include security issues in computer networks, the Public Key Infrastructure ecosystem, key exchange protocols, and security mechanisms and protocols at the application, transport, network and data link layers. It will also discuss up-to-date development in the field of network security. Prereq: CS 270 or EE 287 or consent of the instructor.

CS 575 MODELS OF COMPUTATION. (3)

The formal study of computation, including computability and computation with limited resources. Church's thesis and models of computation. Formal languages and machines as recognizers of languages. The Chomsky Hierarchy of language types. Topics may include Turing machines or other basic models of computation; decidability and undecidability; basic complexity theory; finite automata and regular languages; pushdown automata and context-free languages. The course will cover primarily theory, including assignments that utilize concepts covered in lectures. Prereq: CS 375 and engineering standing, or consent of instructor.

CS 585 INTERMEDIATE TOPICS IN COMPUTER SCIENCE (Subtitle required). (3)

Topics to be selected by staff. May be repeated to a maximum of six credits, but only three credits may be earned by a student under the same topic. Prereq: Restricted to computer science and electrical engineering majors. Others by permission.

CS 587 ADVANCED EMBEDDED SYSTEMS. (3)

An advanced course in the design of embedded systems using state-of-the art microcontroller hardware and software development tools. Topics include architecture support for real-time operating systems, language support for embedded and real-time processing, embedded and wireless networking. Prereq: EE/CPE 580 and engineering standing or consent of instructor. (Same as EE/CPE 587.)

CS 610 MASTER'S PROJECT. (3)

Design and implementation of a large computing project under the supervision of a member of the graduate faculty. Prereq: Satisfactory completion of the departmental foundational examinations.

CS 611 RESEARCH IN COMPUTER SCIENCE. (1-9)

Doctoral students conduct research work in computer science under supervision of a faculty member from the Department of Computer Science. May be repeated to a maximum of 4 semesters. Prereq: 36 credit hours of graduate course work in computer science and approval of the Departmental Committee on Higher Degrees.

CS 612 INDEPENDENT WORK IN COMPUTER SCIENCE. (1-3)

Reading course for graduate students in computer science. May be repeated to a maximum of nine credits. Prereq: Overall standing of 3.0, and consent of instructor.

CS 616 SOFTWARE ENGINEERING. (3)

This course provides an overview of the software engineering discipline: software requirements, software design, software construction, software management, and software quality. Testing and validation techniques will be emphasized throughout the course. Programs and program fragments will be developed and studied throughout the course to illustrate specific problems encountered in the lifecycle development of software systems. Prereq: At least nine hours of graduate computer science courses.

CS 617 REQUIREMENTS ENGINEERING. (3)

The course examines the requirements phase of the Systems Engineering and Software Engineering lifecycles in detail. Topics include: requirements elicitation, requirements specification, and requirements analysis. Verification and validation techniques are emphasized throughout the course. Students work in small groups to research and present a related topic. Prereq: Nine hours of graduate study.

CS 618 SOFTWARE DESIGN. (3)

This course provides an overview of the software design field: software design overview, software design process, a survey of software design method (such as structured design methods, object-oriented design methods, concurrent design methods), design reviews, as well as discussing current topics such as aspect-oriented programming, refactoring, and design patterns. Testing and validation techniques are emphasized through the course. Program designs are developed and validated throughout the course. Readings and summaries of current and seminal journal papers and texts are required. Prereq: Nine hours of graduate study.

KEY: # = new course   * = course changed   † = course dropped

CS   Computer Science

CS 621 PARALLEL AND DISTRIBUTED COMPUTING.   (3)
This course provides graduate students in computer science and in other fields of science and engineering with experience of parallel and distributed computing. It gives an overview of parallel and distributed computers, and parallel computation. The course addresses architectures, languages, environments, communications, and parallel programming. Emphasis on understanding parallel and distributed computers and portable parallel programming with MPI. Prereq: Two 500 level CS courses, or consent of the instructor.

CS 622 MATRIX THEORY AND NUMERICAL LINEAR ALGEBRA II.   (3)
Numerical solution of matrix eigenvalue problems and applications of eigenvalues. Normal forms of Jordan and Schur. Vector and matrix norms. Perturbation theory and bounds for eigenvalues. Stable matrices and Lyapunov theorems. Nonnegative matrices. Iterative methods for solving large sparse linear systems. Prereq: MA 522 or equivalent. (Same as MA 622.)

CS 623 PARALLEL ITERATIVE COMPUTING.   (3)
The course will present advanced computational science techniques needed to support large scale engineering and scientific computations. Emphasis on iterative methods for solving large sparse linear systems and parallel implementations of iterative techniques. Prereq: CS 537 or consent of the instructor.

#CS 626 LARGE SCALE DATA SCIENCE.   (3)
This course will offer an opportunity for students to learn big data techniques and apply them to tackle real-world data science challenges (e.g., processing, storing, querying, exploring, and mining big data). Topics include big data systems and programming models, parallel computing framework, scalable data management and processing solutions, scalable data mining techniques for large datasets, and advanced applications. Prereq: CS 505 or consent of the instructor.

CS 630 FREE-FORM SOLID MODELING.   (3)
This course covers the path from a conceptual vision of a shape to a concrete computer-based description that is suitable for manufacturing. It covers various solids modeling techniques, including volume representations, boundary representations, instantiation and Boolean combinations of shapes, and procedural generation such as sweeps. It discusses effective data structures and consistent and unambiguous part description formats to transfer a shape from a designer to a fabrication house, as well as problems with maintaining unambiguous topology in the presence of finite-precision geometry. Prereq: CS 535 or consent of instructor.

CS 631 COMPUTER-AIDED GEOMETRIC DESIGN.   (3)
Overview of current concepts and issues in CAGD with emphasis on free-form surface design; mathematics of free-form curve and surface representations, including Coons patches, Gregory patches, Bezier method, B-splines, NURBS, triangular interpolants, and their geometric consequences; creating objects with smooth surfaces, covering assembling spline patches, geometric and parametric continuity, texture mapping onto complex shapes, subdivision surfaces, surface evolution, and global optimization. Prereq: CS 535 and CS 321, or consent of instructor.

CS 633 3D COMPUTER ANIMATION.   (3)
This course covers the underlying principles and techniques of 3D computer animation. The topics covered include (1) modeling: the process of building the forms that will be animated, (2) rendering: the process of defining how the final picture in the model will look, (3) animation techniques: the process of creating in-between frames and keyframes, (4) compositing and special effects: the process of assembling various pieces of an image to get special two-dimensional effects, and (5) recording: the principles and techniques involved in putting animation frames onto film or video. Prereq: CS 335 or CS 535, or consent of instructor.

CS 634 MULTIMEDIA SYSTEMS.   (3)
This course covers fundamental techniques in multimedia systems for capturing, managing, accessing and delivering digital media over local, wide-area and wireless network technology. The core topics will emphasize the digital media (images, video, audio) and the algorithms to generate, store, access and process it. Network concepts will be presented at a high level only. Prereq: CS 335 or consent of instructor.

CS 635 IMAGE PROCESSING.   (3)
The course outlines applications of image processing and addresses basic operations involved. Topics covered include image perception, transforms, compression enhancement, restoration, segmentation, and matching. Prereq: Graduate standing and consent of instructor. (Same as EE 635.)

KEY: **# = new course**   *** = course changed**   **† = course dropped**

CS   Computer Science

CS 636 COMPUTER VISION. (3)
This course covers digital image processing as well as advanced topics in computer vision. Initial topics include image formation, digital filtering, sensor modeling and feature detection techniques. The course will discuss how these algorithms are used to address general computer vision problems including three-dimensional reconstruction, scene understanding, object recognition, and motion analysis. Prereq: CS 536 or consent of instructor.

CS 637 EXPLORING VIRTUAL WORLDS. (3)
This course covers a mixture of core techniques related to systems for constructing and modeling virtual environments, such as model-building, image-based rendering, head-mounted hardware, stereo image generation, head-tracking, and immersive display technology. The core topics will be presented using textbooks and papers from the current literature. A substantial group project will provide hands-on experience with the concepts, algorithms and technology. Prereq: CS 335 and CS 635.

CS 642 DISCRETE EVENT SYSTEMS. (3)
The objective of the course is to prepare students for research in the field of supervisory control of discrete event systems (DES's). Logical models, supervising control. Stability and optimal control of DES, complexity analysis and other related research areas will be covered. Prereq: Graduate standing or consent of instructor. (Same as EE 642.)

CS 655 PROGRAMMING LANGUAGES. (3)
Overview of programming-language styles: imperative, functional, declarative, object-oriented, concurrent, simulation, glue. Non-local referencing environments, combinatorial control structures (backtracking, coroutines), higher-order types, lazy/eager evaluation. This course looks at features, not complete languages, touching on such languages as Ada, CLU, FP, Haskell, Icon, Lisp, ML, Modula-2, Modula-3, Pascal, Post, Prolog, Russell, CSim, Simula-67, and Smalltalk-80. Students will not become proficient in any of these languages, but rather will learn what contributions each has made to the state of the art in language design. Compiler-construction issues will be touched on only in passing. Prereq: CS 450G or consent of instructor.

CS 660 TOPICS IN ARTIFICIAL INTELLIGENCE (Subtitle required). (3)
Advanced topics chosen from the following: knowledge representation, knowledge acquisition, problem solving, very high-level programming languages, expert systems, intelligent and deductive databases, automated theorem proving. May be repeated to a maximum of six credits, but only three credits may be earned under the same topic. Prereq: CS 505 and CS 560 or consent of instructor.

CS 663 ARTIFICIAL INTELLIGENCE. (3)
Overview of modern artificial intelligence. Covers topics such as searching and game trees, knowledge representation techniques, methods to represent uncertain information and to reason about it, reasoning about action and planning, expert systems, machine learning and neural networks. Prereq: CS 555 or consent of instructor.

CS 670 DISTRIBUTED OPERATING SYSTEM THEORY. (3)
This course covers advanced distributed operating system algorithms and theory. Topics such as distributed mutual exclusion, distributed event ordering, distributed deadlock detection/avoidance, agreement protocols, consistent global snapshot collection, stable predicate detection, failure recovery, faulty-tolerant consensus, leader election, process groups and group communication. Case studies of distributed operating systems such as LOCUS, Grapevine, V System, ISIS, Amoeba, Sprite, and Mach will be used as illustrations of the above algorithms. Prereq: CS 570 or consent of instructor.

CS 671 ADVANCED COMPUTER NETWORKS. (3)
This course is intended to provide students with a solid understanding of the state of the art in computer network systems and protocols. Topics are covered in some depth, including both abstract and concrete aspects. The course begins with a study of implementations of the current Internet Protocols (TCP, UDP and IP); this provides a concrete backdrop for the rest of the course. The emphasis is on learning by doing, with programming and other hands-on assignments associated with most topics. Prereq: CS 571 or consent of instructor.

CS 673 ERROR CORRECTING CODES. (3)
The problem of correct transmission of data in a noisy environment. The design and analysis of codes that efficiently (in terms of data rate and encryption and decryption speed) correct errors. Linear and nonlinear block codes, general encoding and decoding techniques, fundamental bounds, dual codes, cyclic codes. Specific codes will be studied, including Hamming, BCH, Reed-Muller, Reed-Solomon, trellis, and convolutional codes. Prereq: CS 515 or consent of the instructor.

KEY: # = new course   * = course changed   † = course dropped

CS   Computer Science

CS 674 HEURISTIC ALGORITHMS.   (3)

Solving problems that are intractable. Exact techniques such as search integer programming and dynamic programming. Approximation techniques including local search, divide and conquer, and greedy algorithms. Methods based upon natural models such as force-directed iteration, simulated annealing, genetic algorithms, and neural networks. Examples will be selected from active research areas. Prereq: CS 515 or consent of instructor.

CS 675 COMPUTABILITY AND COMPLEXITY.   (3)

The formal study of computation, including computability and computation with limited resources. Church's thesis and models of computation. Topics will include Turing machines or other basic models of computation; reductions; computable and computably enumerable sets; Rice's Theorem; decidability and undecidability; basic complexity theory; NP-completeness and notions of intractability. Additional topics may include primitive recursive functions and Grzegorczyk hierarchy; nondeterminism; the arithmetic hierarchy; formal complexity measures; time and space hierarchy theorems; the polynomial hierarchy and PSPACE; probabilistic complexity classes; circuit complexity. Prereq: CS 575 or consent of instructor.

CS 677 COMPUTATIONAL GEOMETRY.   (3)

Design and analysis of algorithms and data structures for geometric problems. The particular groups of problems addressed include convex hull construction, proximity, Voronoi Diagrams, geometric search, intersection. Prereq: CS 580.

CS 678 CRYPTOGRAPHY.   (3)

The study of security in communications and electronic computing. The encryption of data using public key systems, block ciphers, and stream ciphers. The basic tools for the design and analysis of such systems. Topics may include information theory, authentication, digital signatures, secret sharing schemes, complexity theoretic issues, probabilistic encryption, electronic commerce and others. Prereq: CS 515 or consent of the instructor.

CS 680 SEMINAR IN COMPUTER SCIENCE.   (2)

May be repeated to a maximum of four credits. Prereq: Consent of instructor, or two 500-level computer science courses.

CS 683 FINITE-STATE MACHINES.   (3)

Analysis and synthesis of sequential machines via state-tables and regular expressions. Equivalence, minimization and decomposition of machines. Partitions and structure-theory. Identification and diagnosis of finite-state machines by means of input-output experiments. Linear, finite-memory, and information-lossless machines. Prereq: EE 280.

CS 684 SPECIAL TOPICS IN VISION, GRAPHICS AND MULTIMEDIA (Subtitle required).   (3)

Advanced topics in computer graphics, computer vision, and multimedia systems. Specific topics include but are not limited to: isophotes, volume rendering, displacement mapping, geographic information systems (GIS), remote sensing topics, large scale sensor networks, video and audio encoding, visualization, immersive environments, and multimedia interfaces. May be repeated to a maximum of up to 6 credit hours, with no more than 3 in the same topic. Prereq: Consent of instructor.

CS 685 SPECIAL TOPICS IN COMPUTER SCIENCE (Subtitle required).   (3)

Topics to be selected by staff. May be repeated to a maximum of six credits but only three credits may be earned under the same topic. Prereq: Consent of instructor or two 500-level computer science courses.

CS 686 SPECIAL TOPICS IN THE THEORY OF COMPUTATION (Subtitle required).   (3)

Advanced topics in the theory of computation and the design and analysis of algorithms, including heuristic approaches for algorithm design, parallel computation, flow problems, parallel and concurrent processes and other areas of current research interest. May be repeated to a maximum of six credits but only three credits may be earned under the same topic. Prereq: Consent of instructor or CS 575 and CS 580.

CS 687 SPECIAL TOPICS IN SYSTEMS.   (3)

This course is a special topics course. The topic and syllabus will change each time the course is offered, reflecting the interests of the instructor. Typically the course will survey new research in the topic area but may also look back at canonical and ground breaking work from the past. Example course topics might include things such as web operating systems, global file systems, distributed object-based systems, fault tolerance/distributed check pointing, high-speed networking, network security, active networking, group communication models, compilers for parallel/distributed computing, recent programming languages, and data mining. Prereq: Consent of instructor.

KEY: **# = new course**   *** = course changed**   **† = course dropped**

CS   Computer Science

CS 689 SPECIAL TOPICS IN NUMERICAL AND SCIENTIFIC COMPUTING (Subtitle required).   **(3)**
Advanced topics in numerical analysis, scientific computation, and complexity of continuous problems. Specific topics may include, but are not limited to: iterative methods, advanced parallel algorithms in numerical linear algebra, multivariate function approximation and integration. Prereq: CS 537 or consent of instructor.

CS 690 OPERATING SYSTEMS THEORY.   **(3)**
An advanced study of operating systems theory including cooperating sequential processes, processor scheduling, paging systems, and memory management. Prereq: CS 570.

CS 748 MASTER'S THESIS RESEARCH.   **(0)**
Half-time to full-time work on thesis. May be repeated to a maximum of six semesters. Prereq: All course work toward the degree must be completed.

CS 749 DISSERTATION RESEARCH.   **(0)**
Half-time to full-time work on dissertation. May be repeated to a maximum of six semesters. Prereq: Registration for two full-time semesters of 769 residence credit following the successful completion of the qualifying exams.

CS 767 DISSERTATION RESIDENCY CREDIT.   **(2)**
Residency credit for dissertation research after the qualifying examination. Students may register for this course in the semester of the qualifying examination. A minimum of two semesters are required as well as continuous enrollment (Fall and Spring) until the dissertation is completed and defended.

CS 768 RESIDENCE CREDIT FOR MASTER'S DEGREE.   **(1-6)**
May be repeated to a maximum of 12 hours.

CS 769 RESIDENCE CREDIT FOR DOCTOR'S DEGREE.   **(0-12)**
May be repeated indefinitely.

KEY: **# = new course**   *** = course changed**   **† = course dropped**