

Computer Science (COM S)

Courses primarily for undergraduates:

COM S 101. Orientation.

Cr. R. F.S.

Introduction to the procedures and policies of Iowa State University and the Department of Computer Science, test-outs, honorary societies, etc. Issues relevant to student adjustment to college life will also be discussed. Offered on a satisfactory-fail basis only.

COM S 103. Computer Applications.

Cr. 4. F.S.SS.

Introduction to computer literacy and applications. Applications: Windows, Internet browser/HTML, word processing, spreadsheets, database management and presentation software. Literacy: history of computing, structure of computers, telecommunications, computer ethics, computer crime, and history of programming languages. No prior computer experience necessary. Course is offered online only. Students must attend an orientation session the first week of class.

COM S 104. Introduction to Computers.

(1.5-1) Cr. 2. F.S.

Offered first 8 weeks and last 8 weeks. Use of personal computer and workstation operating systems and beginning programming. Project-oriented approach to computer operation and programming, including use of tools to aid in programming. Topics from computer history, using basic Windows and Unix tools, program structure, expression, variables, decision and logic, and iteration. No prior computer experience necessary.

COM S 107. Applied Computer Programming.

(3-0) Cr. 3. F.S.

Introduction to computer programming for non-majors using a language such as the Visual Basic language. Basics of good programming and algorithm development. Graphical user interfaces.

COM S 201. Computer Programming in COBOL.

(3-0) Cr. 3. SS.

Computer programming in COBOL. Emphasis on the design, writing, debugging, and testing of business applications programs in a transaction-oriented environment.

COM S 203. Careers in Computer Science.

Cr. R. F.S.

Computer science as a profession. Introduction to career fields open to computer science majors. Relationship of coursework to careers. Presentations by computer science professionals. Offered on a satisfactory-fail basis only.

COM S 207. Fundamentals of Computer Programming.

(Cross-listed with MIS). (3-1) Cr. 3. F.S. *Prereq: MATH 150 or placement into MATH 140/MATH 141/MATH 142 or higher*

An introduction to computer programming using an object-oriented programming language. Emphasis on the basics of good programming techniques and style. Extensive practice in designing, implementing, and debugging small programs. Use of abstract data types. Interactive and file I/O. Exceptions/error-handling. This course is not designed for computer science, software engineering, and computer engineering majors. Credit may not be applied toward graduation for both Com S 207/MIS 207 and Com S 227.

COM S 208. Intermediate Computer Programming.

(3-1) Cr. 3. S. *Prereq: MIS/COM S 207, credit or enrollment in MATH 151, MATH 160, or MATH 165*

Intermediate-level programming techniques. Emphasis on designing, writing, testing, debugging, and documenting medium-sized programs. Data structures and their uses. Dynamic memory usage. Inheritance and polymorphism. Algorithm design and efficiency: recursion, searching, and sorting. Event-driven and GUI programming. The software development process. This course is not designed for computer science, software engineering and computer engineering majors. Credit may not be applied toward the major in computer science, software engineering, or computer engineering.

COM S 227. Introduction to Object-oriented Programming.

(3-2) Cr. 4. F.S.

An introduction to object-oriented design and programming techniques. Symbolic and numerical computation. Recursion and iteration. Modularity procedural and data abstraction, specifications and subtyping. Object-oriented techniques. Imperative programming. Emphasis on principles of programming and object-oriented design through extensive practice in design, writing, running, debugging, and reasoning about programs. This course is designed for majors. Credit may not be applied toward graduation for both Com S 207 and 227.

COM S 228. Introduction to Data Structures.

(3-1) Cr. 3. F.S. *Prereq: C- or better in 227, credit or enrollment in MATH 165*

An object-oriented approach to data structures and algorithms. Object-oriented analysis, design, and programming, with emphasis on data abstraction, inheritance and subtype polymorphism. Abstract data type specification and correctness. Collections and associated algorithms, such as stacks, queues, lists, trees. Searching and sorting algorithms. Graphs. Data on secondary storage. Analysis of algorithms. Emphasis on object-oriented design, writing and documenting medium-sized programs. This course is designed for majors.

COM S 229. Advanced Programming Techniques.

(3-0) Cr. 3. F.S. *Prereq: COM S 228, credit or enrollment in MATH 166*

Object-oriented programming experience using a language suitable for exploring advanced topics in programming. Topics include memory management, parameter passing, inheritance, compiling, debugging, and maintaining programs. Significant programming projects.

COM S 252. Linux Operating System Essentials.

(3-0) Cr. 3. F. *Prereq: COM S 107 or COM S 207 or COM S 227*

Introduction to installation, utilization, and administration of Linux systems. Topics include open-source software, package installation and management, shell programming and command-line utilities, process and service management, account management, network configuration, file sharing, interoperation with other computers and operating systems, automation, and system security.

COM S 290. Independent Study.

Cr. arr. F.S. *Prereq: Permission of instructor*

Offered on a satisfactory-fail basis only.

COM S 290H. Independent Study: Honors.

Cr. arr. F.S. *Prereq: Permission of instructor*

Offered on a satisfactory-fail basis only.

COM S 309. Software Development Practices.

(3-1) Cr. 3. F.S. *Prereq: COM S 228 with C- or better*

A practical introduction to methods for managing software development. Process models, requirements analysis, structured and object-oriented design, coding, testing, maintenance, cost and schedule estimation, metrics. Programming projects. Nonmajor graduate credit.

COM S 311. Design and Analysis of Algorithms.

(3-1) Cr. 3. F.S. *Prereq: COM S 228 with C- or better, MATH 166, ENGL 250, and COM S 330 or CPR E 310*

Basic techniques for design and analysis of efficient algorithms. Sorting, searching, graph algorithms, computational geometry, string processing and NP-completeness. Design techniques such as dynamic programming and the greedy method. Asymptotic, worst-case, average-case and amortized analyses. Data structures including heaps, hash tables, binary search trees and red-black trees. Programming projects. Nonmajor graduate credit.

COM S 319. Software Construction and User Interfaces.

(Cross-listed with S E). (3-0) Cr. 3. F. *Prereq: COM S 228*

Basic theory of grammars, parsing. Language paradigms. State transition and table-based software design. Review of principles of object orientation, object oriented analysis using UML. Frameworks and APIs. User interface architecture, evaluation of user interface. Design of windows, menus, and commands. Introduction to formal specification and model-based software design. Introduction to domain-specific software engineering. Nonmajor graduate credit.

COM S 321. Introduction to Computer Architecture and Machine-Level Programming.

(3-1) Cr. 3. F.S. *Prereq: C- or higher in COM S 228, CPR E 281 and ENGL 250*

Introduction to computer architecture and organization. Emphasis on evaluation of performance, instruction set architecture, datapath and control, memory-hierarchy design, and pipelining. Assembly language on a simulator. Nonmajor graduate credit.

COM S 330. Discrete Computational Structures.

(3-1) Cr. 3. F.S. *Prereq:* C- or higher in MATH 166 and ENGL 250

Concepts in discrete mathematics as applied to computer science. Logic, proof techniques, set theory, relations, graphs, combinatorics, discrete probability and number theory. Nonmajor graduate credit.

COM S 331. Theory of Computing.

(Cross-listed with LING). (3-1) Cr. 3. F.S. *Prereq:* C- or higher in COM S228, C- or higher in COM S 330 or CPR E 310, C- or higher in MATH 166, and ENGL 250

Models of computation: finite state automata, pushdown automata and Turing machines. Study of grammars and their relation to automata. Limits of digital computation, unsolvability and Church-Turing thesis. Chomsky hierarchy and relations between classes of languages. Nonmajor graduate credit.

COM S 336. Introduction to Computer Graphics.

(3-0) Cr. 3. F. *Prereq:* COM S 229, CoReq MATH 307 or MATH 317

Basic algorithms, design, and programming of interactive computer graphics systems and hardware. Topics include 2D and 3D transformations, 3D viewing, visible surface algorithms, collision detection, illumination models, shading, ray tracing, shadows, transparency and texture mapping.

COM S 342. Principles of Programming Languages.

(Cross-listed with S E). (3-1) Cr. 3. F.S. *Prereq:* C- or better in COM S 228, COM S 330 or CPRE 310

Study of concepts in programming languages and major programming paradigms, especially functional programming. Special emphasis on design tradeoffs that enable students to make sound choices of programming languages for a given software development task. Programming projects. Nonmajor graduate credit.

COM S 350. Number Theory.

(Cross-listed with MATH). (3-0) Cr. 3. S. *Prereq:* MATH 166

Divisibility, integer representations, primes and divisors, linear diophantine equations, congruences, and multiplicative functions. Applications to cryptography. Nonmajor graduate credit.

COM S 352. Introduction to Operating Systems.

(3-1) Cr. 3. F.S. *Prereq:* COM S 229, and COM S 321; ENGL 250

Survey of operating system issues. Introduction to hardware and software components including: processors, peripherals, interrupts, management of processes, threads and memory, deadlocks, file systems, protection, virtual machines and system organization, and introduction to distributed operating systems. Programming projects. Nonmajor graduate credit.

COM S 362. Object-Oriented Analysis and Design.

(3-0) Cr. 3. F.S. *Prereq:* COM S 228 with C- or better, ENGL 250

Object-oriented requirements analysis and systems design. Design notations such as the Unified Modeling Language. Design Patterns. Group design and programming with large programming projects. Nonmajor graduate credit.

COM S 363. Introduction to Database Management Systems.

(3-0) Cr. 3. F.S. *Prereq:* COM S 228 with C- or better, ENGL 250

Relational, object-oriented, and semistructured data models and query languages. SQL, ODMG, and XML standards. Database design using entity-relationship model, data dependencies and object definition language. Application development in SQL-like languages and general purpose host languages with application program interfaces. Information integration using data warehouses, mediators and wrappers. Programming Projects. Nonmajor graduate credit.

COM S 398. Cooperative Education.

Cr. R. *Prereq:* Permission of department chair

Required of all cooperative students. Students must register for this course prior to commencing each work period.

COM S 401. Projects in Computing Applications.

(2-2) Cr. 3. F. *Prereq:* ENGL 250, SP CM 212, COM S 309, and either COM S 362 or COM S 363

Applications of software development methods (requirements collection and analysis, software design, project management, documentation and testing), programming techniques, database designs and administration, network application programming to solve computing needs in business settings. A study of practical applications of emerging technologies in computing. Emphasis on semester-long team programming projects. Lab assignments. Oral and written reports. Nonmajor graduate credit.

COM S 409. Software Requirements Engineering.

(Dual-listed with COM S 509). (Cross-listed with S E). (3-0) Cr. 3. F. *Prereq:* COM S 309, ENGL 250, SP CM 212

The requirements engineering process, including identification of stakeholders, requirements elicitation techniques such as interviews and prototyping, analysis fundamentals, requirements specification, and validation. Use of Models: State-oriented, Function-oriented, and Object-oriented. Documentation for Software Requirements. Informal, semi-formal, and formal representations. Structural, informational, and behavioral requirements. Non-functional requirements. Use of requirements repositories to manage and track requirements through the life cycle. Case studies, software projects, written reports, and oral presentations will be required. Nonmajor graduate credit.

COM S 412. Formal Aspects of Specification and Verification.

(Cross-listed with CPR E, S E). (3-0) Cr. 3. *Prereq:* S E 319, COM S 309

Introduction to propositional/predicate/temporal logic, program verification using theorem proving, model-based verification using model checking, and tools for verification. Nonmajor graduate credit.

COM S 414. Gerontechnology in Smart Home Environments.

(Dual-listed with COM S 514). (3-0) Cr. 3. F. *Prereq:* Com S 227 or (Com S 207 or Geron 377 or ArtGr 271) or equivalent.

An interdisciplinary course designed for students who are interested in assistive technology, pervasive computing, mobile computing and principles of universal and inclusive design for end users, in particular, the elderly population. Students will work in semester-long projects as interdisciplinary teams to apply knowledge obtained from lectures and mutual presentations. For graduate credit students are required to submit a research report and give an oral presentation.

COM S 415. Gerontechnology in Smart Home Environments.

(Cross-listed with GERON). (3-0) Cr. 3. F. *Prereq:* Com S 227 or (Com S 207 or Geron 377)

An interdisciplinary course designed for students who are interested in assistive technology, pervasive computing, mobile computing and principles of universal software design for end users, in particular the elderly population. Students will have the chance to learn both about the theories and principles about aging and assistive technology, as well as to engage in the practical semester-long project while working with students from other disciplines. Nonmajor graduate credit.

COM S 417. Software Testing.

(Cross-listed with S E). (3-0) Cr. 3. S. *Prereq:* COM S 309, COM S 319, ENGL 250, SP CM 212

Comprehensive study of software testing, principles, methodologies, management strategies and techniques. Test models, test design techniques (black box and white box testing techniques), integration, regression, system testing methods, and software testing tools. Nonmajor graduate credit.

COM S 418. Introduction to Computational Geometry.

(Dual-listed with COM S 518). (3-0) Cr. 3. Alt. S., offered 2013. *Prereq:* COM S 311 or permission of instructor

Introduction to data structures, algorithms, and analysis techniques for computational problems that involve geometry. Line segment intersection, polygon triangulation, 2D linear programming, range queries, point location, arrangements and duality, Voronoi diagrams and Delaunay triangulation, convex hulls, robot motion planning, visibility graphs. Other selected topics. Programming assignments. Nonmajor graduate credit.

COM S 421. Logic for Mathematics and Computer Science.

(Cross-listed with MATH). (3-0) Cr. 3. S. *Prereq:* MATH 301 or MATH 317 or COM S 330

Propositional and predicate logic. Topics selected from Horn logic, equational logic, resolution and unification, foundations of logic programming, reasoning about programs, program specification and verification, model checking and binary decision diagrams, temporal logic and modal logic. Nonmajor graduate credit.

COM S 425. High Performance Computing for Scientific and Engineering Applications.

(Cross-listed with CPR E). (3-1) Cr. 3. S. *Prereq:* COM S 311, COM S 330, ENGL 250, SP CM 212

Introduction to high performance computing platforms including parallel computers and workstation clusters. Discussion of parallel architectures, performance, programming models, and software development issues. Sample applications from science and engineering. Practical issues in high performance computing will be emphasized via a number of programming projects using a variety of programming models and case studies. Oral and written reports. Nonmajor graduate credit.

COM S 426. Introduction to Parallel Algorithms and Programming.

(Dual-listed with COM S 526). (Cross-listed with CPR E). (3-2) Cr. 4. F. *Prereq:* CPR E 308 or COM S 321, COM S 311

Models of parallel computation, performance measures, basic parallel constructs and communication primitives, parallel programming using MPI, parallel algorithms for selected problems including sorting, matrix, tree and graph problems, fast Fourier transforms. Nonmajor graduate credit.

COM S 430. Advanced Programming Tools.

(3-1) Cr. 3. F. *Prereq:* COM S 311, COM S 362 or COM S 363, ENGL 250, SP CM 212

Topics in advanced programming techniques and tools widely used by industry (e.g., event-driven programming and graphical user interfaces, standard libraries, client/server architectures and techniques for distributed applications). Emphasis on programming projects in a modern integrated development environment. Oral and written reports. Nonmajor graduate credit.

COM S 433. Computational Models of Nanoscale Self-Assembly.

(Dual-listed with COM S 533). (3-0) Cr. 3. S. *Prereq:* C- or higher in COM S 331 or consent of the instructor

Modeling and analysis of natural and engineered systems that spontaneously assemble themselves from small components. Topics include biomolecular self-assembly, tile assembly models, computation via self-assembly, distributed folding, origami models, and self-repair. Emphasis on mathematical methods of describing, simulating, programming, and verifying the behaviors of self-assembling systems. Graduate credit requires a written or oral report on current research. Nonmajor graduate credit.

COM S 437. Computer Game and Media Programming.

(3-0) Cr. 3. S. *Prereq:* COM S 362, COM S 321, COM S 336

Students will learn video game programming using current game engine interfaces with real hardware. Particular attention is paid to the console architecture, development environment, tool chains, 2D graphics, 3D graphics, controllers, memory management, and audio systems. Students will complete the course by writing a simple game that runs on console hardware. Nonmajor graduate credit.

COM S 440. Principles and Practice of Compiling.

(Dual-listed with COM S 540). (3-1) Cr. 3. Alt. S., offered 2013. *Prereq:* COM S 331, COM S 342, ENGL 250, SP CM 212

Theory of compiling and implementation issues of programming languages. Programming projects leading to the construction of a compiler. Projects with different difficulty levels will be given for 440 and 540. Topics: lexical, syntax and semantic analyses, syntax-directed translation, runtime environment and library support. Written reports. Nonmajor graduate credit.

COM S 444. Introduction to Bioinformatics.

(Cross-listed with BCB, BCBIO, BIOL, CPR E, GEN). (4-0) Cr. 4. F. *Prereq:* MATH 165 or STAT 401 or equivalent

Broad overview of bioinformatics with a significant problem-solving component, including hands-on practice using computational tools to solve a variety of biological problems. Topics include: database searching, sequence alignment, gene prediction, RNA and protein structure prediction, construction of phylogenetic trees, comparative and functional genomics, systems biology. Nonmajor graduate credit.

COM S 454. Distributed and Network Operating Systems.

(Dual-listed with COM S 554). (Cross-listed with CPR E). (3-1) Cr. 3. Alt. S., offered 2013. *Prereq:* COM S 311, COM S 352

Laboratory course dealing with practical issues of design and implementation of distributed and network operating systems and distributed computing environments (DCE). The client server paradigm, inter-process communications, layered communication protocols, synchronization and concurrency control, and distributed file systems. Graduate credit requires additional in-depth study of advanced operating systems. Written reports.

COM S 455. Simulation: Algorithms and Implementation.

(Dual-listed with COM S 555). (3-0) Cr. 3. Alt. F., offered 2012. *Prereq:* COM S 311 and COM S 330, STAT 330, ENGL 150, SP CM 212

Introduction to discrete-event simulation with a focus on computer science applications, including performance evaluation of networks and distributed systems. Overview of algorithms and data structures necessary to implement simulation software. Discrete and continuous stochastic models, random number generation, elementary statistics, simulation of queuing and inventory systems, Monte Carlo simulation, point and interval parameter estimation. Graduate credit requires additional in-depth study of concepts. Oral and written reports. Nonmajor graduate credit.

COM S 461. Principles and Internals of Database Systems.

(Dual-listed with COM S 561). (3-1) Cr. 3. F. *Prereq:* COM S 311, ENGL 250, SP CM 212 and COM S 363

Models for structured and semistructured data. Algebraic, first order, and user-oriented query languages. Database schema design. Physical storage, access methods, and query processing. Transaction management, concurrency control, and crash recovery. Database security. Information integration using data warehouses, mediators, wrappers, and data mining. Parallel and distributed databases, and special purpose databases. Oral and written reports. Nonmajor graduate credit.

COM S 471. Computational Linear Algebra and Fixed Point Iteration.

(Cross-listed with MATH). (3-0) Cr. 3. Alt. F., offered 2011. S. *Prereq:* Math 265 and either Math 266, or 267; knowledge of a programming language

Computational error, solutions of linear systems, least squares, similarity methods for eigenvalues, solution of nonlinear equations in one and several variables. Nonmajor graduate credit.

COM S 472. Principles of Artificial Intelligence.

(Dual-listed with COM S 572). (3-1) Cr. 3. F. *Prereq:* COM S 311, COM S 330 or CPR E 310, STAT 330, ENGL 250, SP CM 212, COM S 342 or comparable programming experience

Specification, design, implementation, and selected applications of intelligent software agents and multi-agent systems. Computational models of intelligent behavior, including problem solving, knowledge representation, reasoning, planning, decision making, learning, perception, action, communication and interaction. Reactive, deliberative, rational, adaptive, learning and communicative agents and multiagent systems. Artificial intelligence programming. Graduate credit requires a research project and a written report. Oral and written reports. Nonmajor graduate credit.

COM S 474. Elements of Neural Computation.

(3-1) Cr. 3. Alt. F., offered 2012. *Prereq:* COM S 311, COM S 330 or CPR E 310, STAT 330, MATH 165, ENGL 250, SP CM 212, COM S 342 or comparable programming experience

Introduction to theory and applications of neural computation and computational neuroscience. Computational models of neurons and networks of neurons. Neural architectures for associative memory, knowledge representation, inference, pattern classification, function approximation, stochastic search, decision making, and behavior. Neural architectures and algorithms for learning including perceptions, support vector machines, kernel methods, bayesian learning, instance based learning, reinforcement learning, unsupervised learning, and related techniques. Applications in Artificial Intelligence and cognitive and neural modeling. Hands-on experience is emphasized through the use of simulation tools and laboratory projects. Oral and written reports. Nonmajor graduate credit.

COM S 477. Problem Solving Techniques for Applied Computer Science.

(Dual-listed with COM S 577). (3-0) Cr. 3. Alt. F., offered 2012. *Prereq:* COM S 228; COM S 330 or CPR E 310, MATH 166, MATH 307 or MATH 317, or consent of the instructor

Selected topics in applied mathematics and modern heuristics that have found applications in areas such as geometric modeling, graphics, robotics, vision, human machine interface, speech recognition, computer animation, etc. Polynomial interpolation, roots of polynomials, resultants, solution of linear and nonlinear equations, approximation, data fitting, fast Fourier transform, linear programming, nonlinear optimization, Lagrange multipliers, genetic algorithms, integration of ODEs, curves, curvature, Frenet Formulas, cubic splines, and Bezier curves. Programming components. Written report for graduate credit. Nonmajor graduate credit.

COM S 481. Numerical Methods for Differential Equations and Interpolation.

(Cross-listed with MATH). (3-0) Cr. 3. S. *Prereq:* MATH 265 and either MATH 266 or MATH 267; knowledge of a programming language

First order Euler method, high order Runge-Kutta method, and multistep method for solving ordinary differential equations. Finite difference and finite element methods for solving partial differential equations. Local truncation error, stability, and convergence for finite difference method. Numerical solution space, polynomial approximation, and error estimate for finite element method. Nonmajor graduate credit.

COM S 486. Fundamental Concepts in Computer Networking.

(3-0) Cr. 3. S. *Prereq:* COM S 352

An introduction to fundamental concepts in the design and implementation of computer communication in both the wired and wireless networks, their protocols, and applications. Layered network architecture in the Internet, applications, transport, Socket APIs, network, and data link layers and their protocols, multimedia networking, and network security. Nonmajor graduate credit.

COM S 487. Principles of Distributed and Network Programming.
(Dual-listed with COM S 587). (3-0) Cr. 3. F. *Prereq:* Com s 352 or Cpr E 489 or equivalent.

Programming paradigms for building distributed and networking applications, including multithreaded client-server programming, socket programming, distributed object frameworks and programming suites, and web computing and security. Introduction to some on-going research issues in distributed and networking applications, including peer-to-peer computing, multimedia communications, and mobile computing and networking. Graduate credit requires a written report and an oral presentation. Nonmajor graduate credit.

COM S 490. Independent Study.
Cr. arr. Repeatable, maximum of 9 credits. F.S. *Prereq:* 6 credits in computer science, permission of instructor
Offered on a satisfactory-fail basis only. No more than 9 credits of Com S 490 may be counted toward graduation.

COM S 490H. Independent Study: Honors.
Cr. arr. Repeatable, maximum of 9 credits. F.S. *Prereq:* 6 credits in computer science, permission of instructor
Offered on a satisfactory-fail basis only. No more than 9 credits of Com S 490 may be counted toward graduation.

Courses primarily for graduate students, open to qualified undergraduates:

COM S 502. Complex Adaptive Systems Seminar.
(Cross-listed with CAS). (1-0) Cr. 1. F.S. *Prereq:* Admission to CAS minor
Understanding core techniques in artificial life is based on basic readings in complex adaptive systems. Techniques of complex system analysis methods including: evolutionary computation, neural nets, agent based simulations (agent based computational economics). Large-scale simulations are to be emphasized, e.g. power grids, whole ecosystems.

COM S 503. Complex Adaptive Systems Concepts and Techniques.
(Cross-listed with CAS). (3-0) Cr. 3. S. *Prereq:* Admission to CAS minor or related field
Survey of complex systems and their analysis. Examples are drawn from engineering, computer science, biology, economics and physics.

COM S 509. Software Requirements Engineering.
(Dual-listed with COM S 409). (Cross-listed with S E). (3-0) Cr. 3. F. *Prereq:* COM S 309

The requirements engineering process including identification of stakeholders requirements elicitation techniques such as interviews and prototyping, analysis fundamentals, requirements specification, and validation. Use of Models: State-oriented, Function-oriented, and Object-oriented. Documentation for Software Requirements. Informal, semi-formal, and formal representations. Structural, informational, and behavioral requirements. Non-functional requirements. Use of requirements repositories to manage and track requirements through the life cycle. Case studies, software projects, written reports, and oral presentations will be required.

COM S 510. Distributed Software Development.
(3-0) Cr. 3. F.
Team with students at a foreign university to develop a software application. Importance of distributed development. Design for distributed development, effective processes for distributed development, cultural issues in distributed development, organizing for distributed development, communication techniques and skills for distributed development.

COM S 511. Design and Analysis of Algorithms.
(Cross-listed with CPR E). (3-0) Cr. 3. F. *Prereq:* COM S 311
A study of basic algorithm design and analysis techniques. Advanced data structures, amortized analysis and randomized algorithms. Applications to sorting, graphs, and geometry. NP-completeness and approximation algorithms.

COM S 512. Formal Methods in Software Engineering.
(3-0) Cr. 3. S. *Prereq:* COM S 311, COM S 330
A study of formal techniques for model-based specification and verification of software systems. Topics include logics, formalisms, graph theory, numerical computations, algorithms and tools for automatic analysis of systems. Graduate credit requires in-depth study of concepts.

COM S 514. Gerontechnology in Smart Home Environments.
(Dual-listed with COM S 414). (3-0) Cr. 3. F. *Prereq:* Com S 227 or (Com S 207 or Geron 377 or ArtGr 271) or equivalent.

An interdisciplinary course designed for students who are interested in assistive technology, pervasive computing, mobile computing and principles of universal and inclusive design for end users, in particular, the elderly population. Students will work in semester-long projects as interdisciplinary teams to apply knowledge obtained from lectures and mutual presentations. For graduate credit students are required to submit a research report and give an oral presentation.

COM S 515. Software System Safety.
(3-0) Cr. 3. F. *Prereq:* COM S 309 or COM S 311, COM S 342
An introduction to the analysis, design, and testing of software for safety-critical and high-integrity systems. Analysis techniques, formal verification, fault identification and recovery, model checking, and certification issues. Emphasizes a case-based and systematic approach to software's role in safe systems.

COM S 518. Introduction to Computational Geometry.
(Dual-listed with COM S 418). (3-0) Cr. 3. Alt. S., offered 2013. *Prereq:* COM S 311 or permission of instructor

Introduction to data structures, algorithms, and analysis techniques for computational problems that involve geometry. Line segment intersection, polygon triangulation, 2D linear programming, range queries, point location, arrangements and duality, Voronoi diagrams and Delaunay triangulation, convex hulls, robot motion planning, visibility graphs. Other selected topics. Programming assignments. A scholarly report must be submitted for graduate credit.

COM S 525. Numerical Analysis of High Performance Computing.
(Cross-listed with MATH, CPR E). (3-0) Cr. 3. Alt. S., offered 2015. *Prereq:* CPR E 308 or MATH 481; experience in scientific programming; knowledge of FORTRAN or C

Introduction to parallelization techniques and numerical methods for state-of-the-art high performance computers. A major component will be a final project in an area related to each student's research interests.

COM S 526. Introduction to Parallel Algorithms and Programming.
(Dual-listed with COM S 426). (Cross-listed with CPR E). (3-2) Cr. 4. F. *Prereq:* CPR E 308 or COM S 321, COM S 311
Models of parallel computation, performance measures, basic parallel constructs and communication primitives, parallel programming using MPI, parallel algorithms for selected problems including sorting, matrix, tree and graph problems, fast Fourier transforms.

COM S 531. Theory of Computation.
(3-0) Cr. 3. S. *Prereq:* COM S 331
A systematic study of the fundamental models and analytical methods of theoretical computer science. Computability, the Church-Turing thesis, decidable and undecidable problems, and the elements of recursive function theory. Time complexity, logic, Boolean circuits, and NP-completeness. Role of randomness in computation.

COM S 533. Computational Models of Nanoscale Self-Assembly.
(Dual-listed with COM S 433). (3-0) Cr. 3. S. *Prereq:* C- or higher in COM S 331 or consent of the instructor

Modeling and analysis of natural and engineered systems that spontaneously assemble themselves from small components. Topics include biomolecular self-assembly, tile assembly models, computation via self-assembly, distributed folding, origami models, and self-repair. Emphasis on mathematical methods of describing, simulating, programming, and verifying the behaviors of self-assembling systems. Graduate credit requires a written or oral report on current research.

COM S 540. Principles and Practice of Compiling.
(Dual-listed with COM S 440). (3-1) Cr. 3. Alt. S., offered 2013. *Prereq:* COM S 331, COM S 342, ENGL 250, SP CM 212

Theory of compiling and implementation issues of programming languages. Programming projects leading to the construction of a compiler. Projects with different difficulty levels will be given for 440 and 540. Topics: lexical, syntax and semantic analyses, syntax-directed translation, runtime environment and library support. Written reports.

COM S 541. Programming Languages.
(3-1) Cr. 3. F. *Prereq:* COM S 342 or COM S 440
Survey of the goals and problems of language design. Formal and informal studies of a wide variety of programming language features including type systems. Creative use of functional and declarative programming paradigms.

COM S 544. Introduction to Bioinformatics.

(Cross-listed with BCB, CPR E, GDCB). (4-0) Cr. 4. F. *Prereq: MATH 165 or STAT 401 or equivalent*

Broad overview of bioinformatics with a significant problem-solving component, including hands-on practice using computational tools to solve a variety of biological problems. Topics include: database searching, sequence alignment, gene prediction, RNA and protein structure prediction, construction of phylogenetic trees, comparative, functional genomics, and systems biology.

COM S 549. Advanced Algorithms in Computational Biology.

(Cross-listed with CPR E). (3-0) Cr. 3. Alt. S., offered 2012. *Prereq: COM S 311 and either COM S 228 or COM S 208*

Design and analysis of algorithms for applications in computational biology, pairwise and multiple sequence alignments, approximation algorithms, string algorithms including in-depth coverage of suffix trees, semi-numerical string algorithms, algorithms for selected problems in fragment assembly, phylogenetic trees and protein folding. No background in biology is assumed. Also useful as an advanced algorithms course in string processing.

COM S 550. Evolutionary Problems for Computational Biologists.

(3-0) Cr. 3. Alt. F., offered 2011. *Prereq: COM S 311 and some knowledge of programming*

Discussion and analysis of basic evolutionary principles and the necessary knowledge in computational biology to solve real world problems. Topics include character and distance based methods, phylogenetic tree distances, and consensus methods, and approaches to extract the necessary information from sequence-databases to build phylogenetic trees.

COM S 551. Computational Techniques for Genome Assembly and Analysis.

(3-0) Cr. 3. Alt. F., offered 2011. *Prereq: COM S 311 and some knowledge of programming*

Huang. Introduction to practical sequence assembly and comparison techniques. Topics include global alignment, local alignment, overlapping alignment, banded alignment, linear-space alignment, word hashing, DNA-protein alignment, DNA-cDNA alignment, comparison of two sets of sequences, construction of contigs, and generation of consensus sequences. Focus on development of sequence assembly and comparison programs.

COM S 552. Principles of Operating Systems.

(3-0) Cr. 3. Alt. S., offered 2012. *Prereq: COM S 352*

A comparative study of high-level language facilities for process synchronization and communication. Formal analysis of deadlock, concurrency control and recovery. Protection issues including capability-based systems, access and flow control, encryption, and authentication. Additional topics chosen from distributed operating systems, soft real-time operating systems, and advanced security issues.

COM S 554. Distributed and Network Operating Systems.

(Dual-listed with COM S 454). (Cross-listed with CPR E). (3-1) Cr. 3. Alt. S., offered 2013. *Prereq: COM S 311, COM S 352*

Laboratory course dealing with practical issues of design and implementation of distributed and network operating systems and distributed computing environments (DCE). The client server paradigm, inter-process communications, layered communication protocols, synchronization and concurrency control, and distributed file systems. Graduate credit requires additional in-depth study of advanced operating systems. Written reports.

COM S 555. Simulation: Algorithms and Implementation.

(Dual-listed with COM S 455). (3-0) Cr. 3. Alt. F., offered 2012. *Prereq: COM S 311 and COM S 330, STAT 330*

Introduction to discrete-event simulation with a focus on computer science applications, including performance evaluation of networks and distributed systems. Overview of algorithms and data structures necessary to implement simulation software. Discrete and continuous stochastic models, random number generation, elementary statistics, simulation of queuing and inventory systems, Monte Carlo simulation, point and interval parameter estimation. Graduate credit requires additional in-depth study of concepts. Oral and written reports.

COM S 556. Analysis Algorithms for Stochastic Models.

(3-0) Cr. 3. Alt. S., offered 2013. *Prereq: COM S 331, MATH 307, and STAT 330*

Introduction to the use of stochastic models to study complex systems, including network communication and distributed systems. Data structures and algorithms for analyzing discrete-state models expressed in high-level formalisms. State space and reachability graph construction, model checking, Markov chain construction and numerical solution, computation of performance measures, product-form models, approximations, and advanced techniques.

COM S 557. Computer Graphics and Geometric Modeling.

(Cross-listed with CPR E, M E). (3-0) Cr. 3. F.S. *Prereq: ME 421, programming experience in C*

Fundamentals of computer graphics technology. Data structures. Parametric curve and surface modeling. Solid model representations. Applications in engineering design, analysis, and manufacturing.

COM S 558. Introduction to the 3D Visualization of Scientific Data.

(Cross-listed with HCI, GEOL). (2-2) Cr. 3. Alt. S., offered 2013. *Prereq: Graduate-student standing in the mathematical or natural sciences*

Introduction to visualizing scientific information with 3D computer graphics and their foundation in human perception. Overview of different visualization techniques and examples of 3D visualization projects from different disciplines (natural sciences, medicine, engineering). Class project in interactive 3D visualization using the OpenDX, VTK or a similar system.

COM S 561. Principles and Internals of Database Systems.

(Dual-listed with COM S 461). (3-1) Cr. 3. F. *Prereq: Graduate classification*

Models for structured and semistructured data. Algebraic, first order, and user-oriented query languages. Database schema design. Physical storage, access methods, and query processing. Transaction management, concurrency control, and crash recovery. Database security. Information integration using data warehouses, mediators, wrappers, and data mining. Parallel and distributed databases, and special purpose databases. Oral and written reports.

COM S 562. Implementation of Database Systems.

(3-0) Cr. 3. F. *Prereq: COM S 461 or COM S 561*

Implementation topics and projects are chosen from the following: Storage architecture, buffer management and caching, access methods, design, parsing and compilation of query languages and update operations, application programming interfaces (APIs), user interfaces, query optimization and processing, and transaction management for relational, object-oriented, semistructured (XML), and special purpose database models; client-server architectures, metadata and middleware for database integration, web databases.

COM S 567. Bioinformatics I (Fundamentals of Genome Informatics).

(Cross-listed with BCB, CPR E). (3-0) Cr. 3. F. *Prereq: COM S 208; COM S 330; STAT 341; credit or enrollment in BIOL 315, STAT 430*

Biology as an information science. Review of algorithms and information processing. Generative models for sequences. String algorithms. Pairwise sequence alignment. Multiple sequence alignment. Searching sequence databases. Genome sequence assembly.

COM S 568. Bioinformatics II (Advanced Genome Informatics).

(Cross-listed with GDCB, STAT, BCB). (3-0) Cr. 3. S. *Prereq: BCB 567, BBMB 301, BIOL 315, STAT 430, credit or enrollment in GEN 411*

Advanced sequence models. Basic methods in molecular phylogeny. Hidden Markov models. Genome annotation. DNA and protein motifs. Introduction to gene expression analysis.

COM S 569. Bioinformatics III (Structural Genome Informatics).

(Cross-listed with BCB, BBMB, CPR E). (3-0) Cr. 3. F. *Prereq: BCB 567, GEN 411, STAT 430*

Algorithmic and statistical approaches in structural genomics including protein, DNA and RNA structure. Structure determination, refinement, representation, comparison, visualization, and modeling. Analysis and prediction of protein secondary and tertiary structure, disorder, protein cores and surfaces, protein-protein and protein-nucleic acid interactions, protein localization and function.

COM S 570. Bioinformatics IV (Computational Functional Genomics and Systems Biology).

(Cross-listed with CPR E, BCB, GDCB, STAT). (3-0) Cr. 3. S. *Prereq: BCB 567, BIOL 315, COM S 311 and either 208 or 228, GEN 411, STAT 430*

Algorithmic and statistical approaches in computational functional genomics and systems biology. Elements of experiment design. Analysis of high throughput gene expression, proteomics, and other datasets obtained using system-wide measurements. Topological analysis, module discovery, and comparative analysis of gene and protein networks. Modeling, analysis, simulation and inference of transcriptional regulatory modules and networks, protein-protein interaction networks, metabolic networks, cells and systems: Dynamic systems, Boolean, and probabilistic models. Multi-scale, multi-granularity models. Ontology-driven, network based, and probabilistic approaches to information integration.

COM S 572. Principles of Artificial Intelligence.

(Dual-listed with COM S 472). (3-1) Cr. 3. F. *Prereq:* COM S 311, COM S 331, STAT 330, COM S 342 or comparable programming experience
Specification, design, implementation, and selected applications of intelligent software agents and multi-agent systems. Computational models of intelligent behavior, including problem solving, knowledge representation, reasoning, planning, decision making, learning, perception, action, communication and interaction. Reactive, deliberative, rational, adaptive, learning and communicative agents. Artificial intelligence programming. Graduate credit requires a research project and a written report. Oral and written reports.

COM S 573. Machine Learning.

(3-1) Cr. 3. S. *Prereq:* COM S 311, COM S 362, STAT 330
Algorithmic models of learning. Design, analysis, implementation and applications of learning algorithms. Learning of concepts, classification rules, functions, relations, grammars, probability distributions, value functions, models, skills, behaviors and programs. Agents that learn from observation, examples, instruction, induction, deduction, reinforcement and interaction. Computational learning theory. Data mining and knowledge discovery using artificial neural networks, support vector machines, decision trees, Bayesian networks, association rules, dimensionality reduction, feature selection and visualization. Learning from heterogeneous, distributed, dynamic data and knowledge sources. Learning in multi-agent systems. Selected applications in automated knowledge acquisition, pattern recognition, program synthesis, bioinformatics and Internet-based information systems. Oral and written reports.

COM S 574. Intelligent Multiagent Systems.

(3-0) Cr. 3. Alt. F., offered 2011. *Prereq:* STAT 330; COM S 331; COM S 572, COM S 573, COM S 472, or COM S 474
Specification, design, implementation, and applications of multi-agent systems. Intelligent agent architectures; infrastructures, languages and tools for design and implementation of distributed multi-agent systems; Multi-agent organizations, communication, interaction, cooperation, team formation, negotiation, competition, and learning. Selected topics in decision theory, game theory, contract theory, bargaining theory, auction theory, and organizational theory. Selected topics in knowledge representation and ontologies. Agent-based systems and the Semantic Web. Applications in distributed intelligent information networks for information retrieval, information integration, inference, and discovery from heterogeneous, autonomous, distributed, dynamic information sources.

COM S 575. Computational Perception.

(Cross-listed with HCI, CPR E). (3-0) Cr. 3. S. *Prereq:* Graduate standing or permission of instructor
This class covers statistical and algorithmic methods for sensing, recognizing, and interpreting the activities of people by a computer. This semester we will focus on machine perception techniques that facilitate and augment human-computer interaction. The main goal of the class is to introduce computational perception on both theoretical and practical levels. Participation in small groups to design, implement, and evaluate a prototype of a human-computer interaction system that uses one or more of the techniques covered in the lectures.

COM S 577. Problem Solving Techniques for Applied Computer Science.

(Dual-listed with COM S 477). (3-0) Cr. 3. Alt. F., offered 2012. *Prereq:* COM S 228; COM S 330 or CPR E 310, MATH 166, MATH 307 or MATH 317, or consent of the instructor
Selected topics in applied mathematics and modern heuristics that have found applications in areas such as geometric modeling, graphics, robotics, vision, human machine interface, speech recognition, computer animation, etc. Homogeneous coordinates and transformations, perspective projection, quaternions and rotations, polynomial interpolation, roots of polynomials, resultants, solution of linear and nonlinear equations, approximation, data fitting, Fourier series and fast Fourier transform, linear programming, nonlinear optimization, Lagrange multipliers, parametric and algebraic curves, curvature, Frenet formulas, Bezier curves. Programming components. A scholarly report is required for graduate credit.

COM S 581. Computer Systems Architecture.

(Cross-listed with CPR E). (3-0) Cr. 3. F. *Prereq:* CPR E 381
Quantitative principles of computer architecture design, instruction set design, processor architecture: pipelining and superscalar design, instruction level parallelism, memory organization: cache and virtual memory systems, multiprocessor architecture, cache coherency, interconnection networks and message routing, I/O devices and peripherals.

COM S 583. Reconfigurable Computing Systems.

(Cross-listed with CPR E). (3-0) Cr. 3. *Prereq:* Background in computer architecture, design, and organization
Introduction to reconfigurable computing, FPGA technology and architectures, spatial computing architectures such as systolic and bit serial adaptive network architectures, static and dynamic rearrangeable interconnection architectures, processor architectures incorporating reconfigurability.

COM S 586. Computer Network Architectures.

(3-0) Cr. 3. F. *Prereq:* COM S 511, COM S 552 or CPR E 489
Design and implementation of computer communication networks: layered network architectures, local area networks, data link protocols, distributed routing, transport services, network programming interfaces, network applications, error control, flow/congestion control, interconnection of heterogeneous networks, TCP/IP, ATM networks, multimedia communications, IP and application multicast, overlay networks, network security and web computing.

COM S 587. Principles of Distributed and Network Programming.

(Dual-listed with COM S 487). (3-0) Cr. 3. F. *Prereq:* Com s 352 or Cpr E 489 or equivalent.
Programming paradigms for building distributed and networking applications, including multithreaded client-server programming, socket programming, distributed object frameworks and programming suites, and web computing and security. Introduction to some on-going research issues in distributed and networking applications, including peer-to-peer computing, multimedia communications, and mobile computing and networking. Graduate credit requires a written report and an oral presentation. Nonmajor graduate credit.

COM S 590. Special Topics.

Cr. arr. Repeatable. *Prereq:* Permission of instructor
Offered on a satisfactory-fail basis only.

COM S 592. Research Colloquia.

Cr. 1. F.S. *Prereq:* Graduate classification
Attend Computer Science Research Colloquia. Written summary is required. Offered on a satisfactory-fail basis only.

COM S 596. Genomic Data Processing.

(Cross-listed with GDCB, BCB). (3-0) Cr. 3. F. *Prereq:* Some experience in computation
Study the practical aspects of genomic data processing with an emphasis on hands-on projects. Topics include base-calling, sequence cleaning and contaminant removal; fragment assembly procedures and EST clustering methods; genome closure strategies and practices; sequence homology search and function prediction; and annotation and submission of GenBank reports. Next-generation sequencing topics like model genome resequencing, short-read assembly and transcriptome abundance measurement will also be covered.

COM S 598. Graduate Internship.

Cr. R. Repeatable. F.S.SS. *Prereq:* Graduate Classification
Supervised internship working in professional settings appropriate to the student's degree program. Academic work under faculty supervision.

COM S 599. Creative Component.

Cr. 1-3.
Creative component for nonthesis option of Master of Science degree. Offered on a satisfactory-fail basis only.

Courses for graduate students:**COM S 610. Seminar.**

Cr. arr.
Offered on a satisfactory-fail basis only.

COM S 611. Advanced Topics in Analysis of Algorithms.

(3-0) Cr. 3. Repeatable. Alt. S., offered 2013. *Prereq:* COM S 511, COM S 531
Advanced algorithm analysis and design techniques. Topics include graph algorithms, algebraic algorithms, number-theoretic algorithms, randomized and parallel algorithms. Intractable problems and NP-completeness. Advanced data structures.

COM S 612. Distributed Algorithms.

(3-0) Cr. 3. Alt. S., offered 2012. *Prereq:* COM S 511 or COM S 531
The theory of distributed computation. Algorithms, lower bounds and impossibility results. Leader Elections, mutual exclusion, consensus and clock synchronization algorithms. Synchronous, asynchronous and partially synchronous distributed systems models. Shared memory and message passing systems. Fault-tolerance and randomization. Broadcast and multicast. Wait-free object simulations. Distributed shared memory.

COM S 625. Issues in Parallel Programming and Performance.(3-0) Cr. 3. Alt. S., offered 2013. *Prereq: COM S 511*

Parallel solutions of numerical and non-numerical problems, implementation of parallel programs on parallel machines, performance and other computational issues in parallel programming.

COM S 626. Parallel Algorithms for Scientific Applications.(Cross-listed with CPR E). (3-0) Cr. 3. *Prereq: CPR E 526*

Algorithm design for high-performance computing. Parallel algorithms for multidimensional tree data structures, space-filling curves, random number generation, graph partitioning and load balancing. Applications to grid and particle-based methods and computational biology.

COM S 631. Advanced Topics in Computational Complexity.(3-0) Cr. 3. Repeatable. Alt. F., offered 2012. *Prereq: COM S 531*

Advanced study in the quantitative theory of computation. Time and space complexity of algorithmic problems. The structure of P, NP, PH, PSPACE, and other complexity classes, especially with respect to resource-bounded reducibilities and complete problems. Complexity relative to auxiliary information, including oracle computation and relativized classes, randomized algorithms, advice machines, Boolean circuits. Kolmogorov complexity and randomness.

COM S 633. Advanced Topics in Computational Randomness.(3-0) Cr. 3. Repeatable. Alt. F., offered 2011. *Prereq: COM S 531*

Advanced study of the role of randomness in computation. Randomized algorithms, derandomization, and probabilistic complexity classes. Kolmogorov complexity, algorithmic information theory, and algorithmic randomness. Applications chosen from cryptography, interactive proof systems, computational learning, lower bound arguments, mathematical logic, and the organization of complex systems.

COM S 634. Theory of Games, Knowledge and Uncertainty.(3-0) Cr. 3. Alt. S., offered 2013. *Prereq: COM S 330*

Fundamentals of Game Theory: individual decision making, strategic and extensive games, mixed strategies, backward induction, Nash and other equilibrium concepts. Discussion of Auctions and Bargaining. Repeated, Bayesian and evolutionary games. Interactive Epistemology: reasoning about knowledge in multiagent environment, properties of knowledge, agreements, and common knowledge. Reasoning about and representing uncertainty, probabilities, and beliefs. Uncertainty in multiagent environments. Aspects and applications of game theory, knowledge, and uncertainty in other areas, especially Artificial Intelligence and Economics, will be discussed.

COM S 641. Advanced Topics in Programming Language Semantics.(3-0) Cr. 3. Repeatable. Alt. S., offered 2012. *Prereq: COM S 531, COM S 541*

Operational and other mathematical models of programming language semantics. Type systems and their soundness. Applications of semantics on areas such as program correctness, language design or translation.

COM S 652. Advanced Topics in Distributed Operating Systems.(3-0) Cr. 3. Repeatable. Alt. F., offered 2011. *Prereq: COM S 552*

Concepts and techniques for network and distributed operating systems: Communications protocols, processes and threads, name and object management, synchronization, consistency and replications for consistent distributed data, fault tolerance, protection and security, distributed file systems, design of reliable software, performance analysis.

COM S 657. Advanced Topics in Computer Graphics.(3-0) Cr. 3. Alt. F., offered 2012. *Prereq: COM S 228, I E 557/M E 557/CPR E 557/COM S 557*

Modern lighting models: Rendering Equation, Spherical Harmonics, Lafortune, Cook-Torrance. Non-polygonal primitives: volumes, points, particles. Textures: filtering, reflections creation. Graphics hardware: pipeline, performance issues, programmability in vertex and fragment path. Per-pixel lighting. Nonphotorealistic rendering. Radiosity; Ray tracing.

COM S 661. Advanced Topics in Database Systems.(3-0) Cr. 3. Repeatable. Alt. F., offered 2012. *Prereq: COM S 461 or COM S 561*

Advanced topics chosen from the following: database design, data models, query systems, query optimization, incomplete information, logic and databases, multimedia databases; temporal, spatial and belief databases, semistructured data, concurrency control, parallel and distributed databases, information retrieval, data warehouses, wrappers, mediators, and data mining.

COM S 672. Advanced Topics in Computational Models of Learning.(3-0) Cr. 3. Repeatable. Alt. S., offered 2012. *Prereq: COM S 572 or COM S 573 or COM S 472 or COM S 474*

Selected topics in Computational Learning Theory (PAC learning, Sample complexity, VC Dimension, Occam Learning, Boosting, active learning, Kolmogorov Complexity, Learning under helpful distributions, Mistake Bound Analysis). Selected topics in Bayesian and Information Theoretic Models (ML, MAP, MDL, MML). Advanced statistical methods for machine learning. Selected topics in reinforcement learning.

COM S 673. Advanced Topics in Computational Intelligence.(3-0) Cr. 3. Repeatable. Alt. S., offered 2013. *Prereq: COM S 572 or COM S 573 or COM S 472 or COM S 474*

Advanced applications of artificial intelligence in bioinformatics, distributed intelligent information networks and the Semantic Web. Selected topics in distributed learning, incremental learning, multi-task learning, multi-strategy learning; Graphical models, multi-relational learning, and causal inference; statistical natural language processing; modeling the internet and the web; automated scientific discovery; neural and cognitive modeling.

COM S 681. Advanced Topics in Computer Architecture.(Cross-listed with CPR E). (3-0) Cr. 3. Alt. S., offered 2013. *Prereq: CPR E 581.**Repeatable with Instructor permission*

Current topics in computer architecture design and implementation. Advanced pipelining, cache and memory design techniques. Interaction of algorithms with architecture models and implementations. Tradeoffs in architecture models and implementations.

COM S 686. Advanced Topics in High-Speed Networks.(3-0) Cr. 3. Alt. S., offered 2013. *Prereq: COM S 586*

Advanced topics in IP networks and optical networks. QoS routing and scheduling, multicast, multiprotocol label switching (MPLS), traffic engineering. Optical network architectures, routing and wavelength assignment algorithms, optical multicast, traffic grooming, optical burst switching, lightpath protection/restoration schemes, and IP over WDM.

COM S 699. Research.Cr. arr. Repeatable. *Prereq: Approval of instructor*

Offered on a satisfactory-fail basis only.

COM S COM S 512. Formal Aspects of Specification and Verification.(Dual-listed with COM S 412). (Cross-listed with COM S, S E). (3-0) Cr. 3. *Prereq: S E 319, COM S 309*

Introduction to propositional/predicate/temporal logic, program verification using theorem proving, model-based verification using model checking, and tools for verification. Nonmajor graduate credit.