COMPUTING AND INFORMATION SCIENCE

FACULTY AND FOUNDERS OF COMPUTING AND INFORMATION SCIENCE ADMINISTRATION

Robert L. Constable, Dean for Computing and Information Science

John Abowd, Industrial and Labor Relations

Kenneth Birman, Computer Science

Claire Cardie, Computer Science

Terrence Fine, Electrical and Computer Engineering

Geri Gay, Communication, CIS

Tarleton Gillespie, Science and Technology Studies, CIS

Paul Ginsparg, Physics, CIS

Donald Greenberg, Architecture, Johnson Graduate School of Management

John Guckenheimer, Mathematics

Juris Hartmanis, Computer Science

Daniel Huttenlocher, Johnson Graduate School of Management, Computer Science

Jon Kleinberg, Computer Science

Hod Lipson, Mechanical and Aerospace Engineering, CIS

N. David Mermin, Physics

Andrew Pershing, Earth and Atmospheric Sciences, CIS

Stephen Pope, Mechanical and Aerospace Engineering

Mats Rooth, Linguistics, CIS

Fred Schneider, Computer Science

Phoebe Sengers, Science and Technology Studies, CIS

David Shalloway, Molecular Biology and Genetics, CIS

Christine A. Shoemaker, Civil and Environmental Engineering

Buzz Spector, Art Department

Ken Torrance, Mechanical and Aerospace Engineering

Charles Van Loan, Computer Science

MISSION

The faculty of Computing and Information Science (FCIS) offers courses and programs campuswide in various academic disciplines in which computing is integral. The faculty associated with CIS programs hold joint appointments with CIS and another Cornell academic unit. The FCIS has, so far, focused its interdisciplinary efforts in the areas of computational biology, computational science and engineering, digital arts and graphics, and information science. Each area of interest is in a different stage of development, with information science leading the way.

Programs in these areas are administered by CIS. Visit the web site at www.cis.cornell.edu/infoscience/undergraduate.html or go to 303 Upson Hall for more information.

INFORMATION SCIENCE

Information Science at Cornell is a new interdisciplinary program that studies the design and use of information systems in a social context. It integrates the study of three aspects of digital information systems. First, Information Science studies computing systems that provide people with information content; this study overlaps with parts of computer science stressing the design, construction, and use of large information systems such as the World Wide Web, the National Digital Library for science education, and other global information resources. The second aspect of Information Science examines how people engage these informa-tion resources and how information systems can be designed to be integrated into everyday life—human-computer interaction (HCI). This area is also called "human-centered systems" because it is concerned with systems that hundreds of millions of people will use in daily life. The third aspect is the study of social systems and how they interact with computer systems. This area of Information Science deals with understanding how information systems are situated in social, economic, and historical contexts. It explores the economic value of information, the legal constraints on systems, their social impact, and the cultural aspects of their construction. These are synergistic topics, and the next generation of scientists, scholars, business leaders, and government workers will need to understand them and how they relate.

Specific topics emphasized in the Information Science program include information networks; information discovery; knowledge organization; interaction design; interface design and evaluation; collaboration within and across groups, communities, organizations, and society; computational linguistics; computational techniques in the collection, archiving, and analysis of social science data; information privacy; methods of collecting, preserving, and distributing information; information system design; cognition and learning; social informatics; and cultural studies of computation.

An undergraduate concentration or minor in Information Science is currently available in the Colleges of Arts and Sciences; Engineering; Agriculture and Life Sciences (CALS); Human Ecology; and Architecture, Art, and Planning (AAP; available to Architecture, Art, and Planning students); and the School of Industrial and Labor Relations (ILR). The requirements for the concentration/minor are described in a separate section below. (Students in the School of Hotel Administration are directed to the Information Systems concentration within the Bachelor of Science degree.)

COMPUTATIONAL SCIENCE AND ENGINEERING

Another FCIS area of activity is computational science and engineering. Numerous courses are taught throughout the university. Topics include numerical methods, modeling and simulation, and real-time computing and control. The FCIS itself sponsors several "tool-based" short courses for students who anticipate that their studies will have a strong computational component (CIS 401, 402, 403, 404). A course on data structures for computational science (CIS 409) also is offered.

DIGITAL ARTS AND GRAPHICS

The FCIS is working to develop research programs and curriculum in the digital arts. Several courses already exist in this area (ART 372, ART 391, MUSIC 120, COM S 465, 467, 468), and more can be expected in the near future.

RELATIONSHIP WITH COMPUTER SCIENCE

FCIS programs have connections to computer science, the study of computation in all of its forms. Computation is both abstract and physical, both artificial and natural, and its study is a unique combination of fundamental science, applied science, and enabling technology. The curriculum covers the theory of algorithms and computing and its many applications in science, engineering, and business. Students learn the algorithmic method of thinking and how to bring it to bear on a wide range of problems. They also study the elements of computing and information technology such as system design, problem specification, programming, system analysis and evaluation, and complex modeling. Research areas include programming languages, compilers, computing systems, artificial intelligence, natural language processing, computer graphics, computer vision, databases, networks, bioinformatics, the theory of algorithms, scientific computing, and computational logic.

Undergraduate majors in Computer Science are offered in the College of Engineering and the College of Arts and Sciences. A minor in Computer Science is available to undergraduates in the College of Engineering. These programs are administered by the Department of Computer Science, whose professors are members of both the College of Engineering and the College of Arts and Sciences.

There is also a one-year Master of Engineering Program in Computer Science and a Ph.D. program in Computer Science. These degree programs are administered by the Graduate Field of Computer Science.

THE INFORMATION SCIENCE CONCENTRATION/MINOR

A concentration/minor in Information Science is available to students in the Colleges of Arts and Sciences; Engineering; Agriculture and Life Sciences (CALS); Human Ecology; and Architecture, Art, and Planning (AAP; available to Architecture, Art, and Planning students), and the School of Industrial and Labor Relations (ILR). Because of small differences in regulations between the colleges, there are sometimes slight variations in the requirements depending on a student's college and, in a few cases, a student's major. All students interested in pursuing the Information Science concentration/minor must initiate the process by sending an e-mail message with their name, college, year of study (e.g., second-semester sophomore), expected graduation date, and (intended) major to infosci-minor@cs.cornell.edu. Students are also referred to www.cis.cornell.edu/infoscience/concentration.htm for the most up-to-date description of the concentration and its requirements.

Information Science is an interdisciplinary field covering all aspects of digital information. The program has three main areas: human-centered systems, social systems, and information systems. Human-centered systems studies the relationship between humans and information, drawing from human-computer interaction and cognitive science. Social systems examines information in its economic, legal, political, cultural, and social contexts. Information systems studies the computer science problems of representing, storing, manipulating, and using digital information.

The concentration/minor has been designed to ensure that students have substantial grounding in all three of these areas. To this end, the requirements for the undergraduate concentration/minor are as follows. All courses must be chosen from the course lists below. In addition, a letter grade of C or better is required; S-U courses are not allowed.

- **Statistics:** one course.
- **Human-centered systems** (human-computer interaction and cognitive science): two courses (for all colleges except Engineering); one course (Engineering).
- **Social systems** (social, economic, political, cultural, and legal issues): one course.
- **Information systems** (primarily computer science): two courses.
- **Elective**: one additional course from any component area. (Engineering students and all Computer Science majors must select a course from human-centered systems or social systems. Communication majors must select a course outside Communication. Students in other majors should check with their advisers to make sure there are no special departmental restrictions or requirements.)

Statistics

An introductory course that provides a working knowledge of basic probability and statistics and their application to analyzing data occurring in the real world.

Engineering students must take one of the following:

- ENGRD 270 Basic Engineering Probability and Statistics
- CEE 304 Uncertainty Analysis in Engineering
- ECE 310 Introduction to Probability and Random Signals

All other students can meet this requirement with any one of the following:

- MATH 171 Statistical Theory and Application in the Real World
- BTRY 261 Statistical Methods I (also STBTRY 261)
- ENGRD 270 Basic Engineering Probability and Statistics
- CEE 304 Uncertainty Analysis in Engineering
- ECE 310 Introduction to Probability and Random Signals
- ILRST 312 Applied Regression Methods
- PSYCH 350 Statistics and Research Design or equivalent with permission of the Information Science undergraduate program director (send mail to infosci-minor@cs.cornell.edu).

Human-Centered Systems

- COGST 101 Introduction to Cognitive Science
- PSYCH 205 Perception
- PSYCH 214 Cognitive Psychology
- COMM 240 Communication and Information Technology
- COMM 245 Psychology of Social Computing
- PSYCH 280 Introduction to Social Psychology
- PSYCH 342 Human Perception: Applications to Computer Graphics, Art, and Visual Display
- COMM 345 Human-Computer Interaction Design
- PSYCH 347 Psychology of Visual Communications
- PSYCH 380 Social Cognition
- PSYCH 413 Information Processing: Conscious and Unconscious
- PSYCH 416 Modeling Perception and Cognition
- COM S 465 Introduction to Computer Graphics
- COMM 440 Advanced Human-Computer Interaction Design
- COMM 450 Language and Technology

Social Systems

- STS 250 Technology in Society
- STS 292 Inventing an Information Society
- ECON 301 Microeconomics*
- ECON 313 Intermediate Microeconomic Theory*
- SOC 304 Social Networks and Social Processes
- STS 349 Media Technologies
- STS 355 Computers: From Babbage to Gates
- STS 387 The Automatic Lifestyle: Consumer Culture and Technology
- LAW 410 Limits on and Protection of Creative Expression—Copyright Law and Its Close Neighbors
- STS 411 Knowledge, Technology, and Property
- ECON 419 Economic Decisions Under Uncertainty
- COMM 428 Communication Law
- OR&IE 435 Introduction to Game Theory*
- STS 438 Minds, Machines, and Intelligence
- ECON 467 Game Theory*

*Only one of ECON 301 and ECON 313 can be taken for IS credit. Only one of OR&IE 435 and ECON 467 can be taken for IS credit.

Information Systems

- CIS 130 Creating Web Documents*
- COM S 211 Computers and Programming*
- CIS 230 Intermediate Web Design*
- CIS 330 Applied Databases
- LING 424 Computational Linguistics
- CIS 430 Information Discovery
- CIS 431 Web Information Systems
- COM S 432 Introduction to Database Systems
- COM S 465 Computer Graphics I
- LING 474 Introduction to Natural Language Processing
- OR&IE 474 Statistical Data Mining
- COM S 478 Machine Learning
- OR&IE 480 Information Technology
- CIS 501 Software Engineering
- ECE 562 Fundamental Information Theory
- COM S 574 Language Technologies
- COM S 578 Empirical Methods in Machine Learning and Data Mining

*CIS 130 cannot be taken for Information Science credit by Engineering students. Computer Science majors cannot use CIS 130 or CIS 230. COM S 211 cannot be taken for Information Science credit by majors for which it is a required course, e.g., Computer Science (COM S) and Operations Research and Industrial Engineering (OR&IE).

CIS COURSES

Courses offered under the CIS rubric are of particular interest to students in the computing and information sciences.

CIS 130 Introductory Web Programming (also COM S 130)

Fall, summer. 3 credits. No prerequisites. Interactive online media such as the World Wide Web are revolutionizing the way we communicate. This course introduces students having little or no computer background to tools and techniques for creating interactive documents. Questions of both design and technical issues are emphasized. Students must think seriously about digital graphic

impact and must learn how to do some relatively simple programming with a scripting language (such as JavaScript). Topics covered include HTML; JavaScript; interaction techniques (elementary DHTML); ways of coping with slow connections; the incorporation of sound, video, and images in web documents; animation techniques (Flash 5); ethics; and e-commerce.

CIS 191 Media Arts Studio I (also ART 391, THETR 391)
Fall. 3 credits. Prerequisite: one of the following courses: ART 171, THETR 277, 377, MUSIC 120, or equivalent; must be a junior and have permission of the instructor. Lab fee $50.
See ART 391 for description.

CIS 230 Intermediate Web Design (also COM S 230)
Spring. 3 credits. Prerequisite: COM S 130 or equivalent.
Web programming requires the cooperation of two machines: the one in front of the viewer (client) and the one delivering the content (server). COM S 130 concentrates almost exclusively on the client side. The main emphasis in COM S 230 is learning about server side processing. Students begin by looking at interactions with databases, learning about querying both on paper and via SQL, and then, through a succession of projects, learn how to apply this understand-ing to the creation of an interactive data-driven site via the use of an integrated web site development tool such as ColdFusion. Also considered are techniques to enhance security, privacy, and reliability and ways of incorporating other programs. Toward the end of the course, students are shown how these development tools are working. Design issues are emphasized. A major component of the course is the creation of a substantial web site.

CIS 330 Applied Database Systems (also COM S 330)
Spring. 3 credits. Prerequisites: COM S 211/ENGRD 211.
An introduction to modern database systems and three-tier application development using database systems. Concepts covered include the relational model, query languages, data modeling, database tuning, three-tier architectures, Internet data formats and query languages, server- and client-side technologies, decision support systems, and an introduction to data mining. The course is targeted at users of database systems and at application development using database systems and includes several programming projects comprising the design and implementation of a database–backed web site.

CIS 387 The Automatic Lifestyle: Consumer Culture and Technology (also ST&S 387)
Spring. 4 credits.
Our daily lifestyle in consumer culture is intimately intertwined with technology. Industrialized technology makes consumer culture possible, yet at the same time the economic and cultural trends of consumer culture select and shape the kinds of technology that become available. How is our daily lifestyle in consumer culture shaped by technology? How are everyday technologies shaped by the demands of consumer culture? What alternatives do we have? In this course, we synthesize history, sociology, and speculative design to answer these questions.

CIS 401 Introduction to Applied Scientific Computing with MATLAB
Fall. 1 credit. Usually weeks 2–5. Prerequisite: COM S 100 or equivalent programming experience. S-U grades only.
An introduction to the use of MATLAB as an aid to scientific research. The course introduces the basic syntax and features of MATLAB and develops the background necessary for the more specialized courses. The course covers basic MATLAB programming and vectorized operations, data input/output, and simple visualization. The course emphasizes applied issues such as managing large data sets, simulation, and visualization but also introduces fundamental ideas in scientific computing such as floating point arithmetic and algorithm efficiency. Although the course uses MATLAB, the ideas and concepts covered are common to many computational environments.

CIS 402 Scientific Visualization with MATLAB
Fall. 1 credit. Usually weeks 6–10. Prerequisites: COM S 100 or equivalent programming experience; COM S 401/CIS 401 recommended but not required. S-U grades only.
A survey of the advanced visualization features in MATLAB. The course covers MATLAB's "handle graphics" paradigm, specialized graphics routines for vectors and fields, and introduces color mapping, lighting, and new features for controlling object transparency. Although the course is meant to introduce students to the capabilities of the MATLAB system, it also emphasizes the basic goal of visualization: producing an image that effectively communicates a scientific result.

CIS 403 Development of Scientific Computing Programs
Spring. 1 credit. Usually weeks 1–4. Prerequisite: COM S 100 or equivalent programming experience. S-U grades only.
This course is designed for graduate students who, in their research, will develop computer programs to solve scientific or engineering problems (e.g., in Fortran, C, or Java). Approaches and tools are presented that facilitate the development of good software. The course emphasizes the tools available in UNIX and Windows environments. Topics covered include compilers, debuggers, software design, and project management.

CIS 404 Survey and Use of Software Libraries for Scientific Computing
Spring. 1 credit. Usually weeks 5–8. Prerequisites: COM S 100 or equivalent programming experience; COM S 403/CIS 403 recommended but not required. S-U grades only.
Many software packages and code libraries have been developed for the solution of standard problems in scientific computing. Examples of such libraries are LAPACK, IMSL, Numerical Recipes routines, MATLAB functions, and routines available in online repositories such as Netlib. This course discusses how to link to or compile standard library formats and considers the legal and ethical aspects of using other people's code (or having them use yours). The course also surveys some of the standard problems and the available libraries and discusses the issues that arise in their use (e.g., accuracy, robustness, and generality).

CIS 409 Data Structures and Algorithms for Computational Science
Fall. 4 credits. Prerequisite: COM S 211 or equivalent programming experience. Not offered every year.
Covers data structures and algorithms with emphasis on those useful for computational science. This course is intended for students outside of the Department of Computer Science whose work involves a significant amount of computing. Topics include basic data structures as well as more advanced topics. Emphasis is placed on the use of abstract data types and on how best to select appropriate data structures.

CIS 430 Information Discovery (also COM S 430)
Fall. 3 credits. Prerequisite: COM S 211 or equivalent.
This course studies the methods used to search for and discover information in large-scale systems. The emphasis is on information retrieval applied to textual materials, but there is some discussion of other formats. The course includes techniques for searching, browsing and filtering information, and the use of classification systems and thesauruses. The techniques are illustrated with examples from web searching and digital libraries.

CIS 431 Web Information Systems (also COM S 431, formerly CIS/COM S 502)
Spring. 3 credits. Prerequisites: COM S 211 and some familiarity with the technology of web sites.
This course examines the architecture of web information systems such as distributed digital libraries and electronic publishing systems. Many of the topics presented are the subject of current research and development at Cornell, other universities, and in standards organizations such as the World Wide Web Consortium. Course content mixes exploration of current tools for building web information systems such as XML, XSLT, and RDF with broader concepts such as techniques for knowledge representation and description, object models for content representation, and legal and economic impacts of web information. A theme that runs throughout the course is the relationship between traditional information environments, exemplified by libraries, and the distributed information environment of the web.

CIS 490 Independent Reading and Research
Fall, spring. 1–4 credits.
Independent reading and research for undergraduates.

CIS 515 Culture, Law, and Politics of the Internet
Fall. 4 credits.
This course explores the culture, law, and politics of the Internet, everything from "No one will know you are a dog on the Internet" to debates on the constitutionality of the Digital Millennium Copyright Act. Free speech concerns, Internet governance, domain naming, copyright, privacy, and security are highlighted as well as a variety of policy issues such as acceptable use bandwidth usage on campuses, protocols for DMCA compliance, and the balance of classroom and distributed learning.

CIS 518 Computer Animation (also ART 372)
Fall. 4 credits. Prerequisite: COM S/ENGRD 211.

This course introduces students to various advanced animation techniques, including modeling, 2D key cel animation, motion and kinematics, lighting and materials, 3D key frame animation, physically based simulation, special effects, story boarding, and cinematography. The first half of the course consists of lectures/seminars and some didactic project assignments with supplemental reading assignments. The second half consists of a final project in which students may work in groups to create an original work of computer animation.

[CIS 576 Decision Theory I (also ECON 476, 676)
Fall. 4 credits. Prerequisites: mathematical sophistication.
See ECON 476 for description.]

[CIS 577 Decision Theory II (also ECON 477, 677)
Spring. 4 credits. Prerequisites: Mathematical sophistication.
See ECON 476 for description.]

CIS 630 Representing and Accessing Digital Information (also COM S 630)
Fall. 4 credits. Prerequisites: COM S 472 or 478 or 578 or the equivalent.
This course covers the representation, organization, and access of digital information with an emphasis on textual information. Topics include structured and semistructured data, information retrieval, natural language processing, and machine learning, with links to work in databases and data mining, and computational linguistics.

[CIS 685 The Structure of Information Networks (also COM S 685)
Fall. 4 credits. Prerequisite: COM S 482.]

[CIS 750 Evolutionary Computation and Design Automation (also COM S 750, M&AE 650)
Fall. 4 credits. Prerequisite: programming experience or permission of instructor. Not offered every year.]

[CIS 751 Media Research and Critical Design (also COM S 751)
Fall. 4 credits. Prerequisites: graduate standing in COM S or equivalent ability to read technical research papers. Contact instructor if unsure of qualifications. Not offered every year.]

[CIS 752 Seminar on Scholarly Information Architecture
Fall. 3 credits. Prerequisite: concurrent enrollment in COM S 431 or equivalent experience. S-U grades only. Not offered every year.]

COMPUTER SCIENCE

The Department of Computer Science is affiliated with both the College of Arts and Sciences and the College of Engineering. Students in either college may major in computer science. The department is also part of CIS, and its courses are an integral part of its several educational programs.

COM S 099 Fundamental Programming Concepts
Fall, summer. 2 credits. No prerequisites. S-U grades only. Credit cannot be applied toward the Engineering degree. Freshmen only.
This course is designed for students who intend to take COM S 100 but are not adequately prepared for that course. Basic programming concepts and problem analysis are studied. An appropriate high-level programming language is used. Students with previous programming experience and students who do not intend to take COM S 100 should not take this course.

COM S 100 Introduction to Computer Programming
Fall, spring, summer. 4 credits.
An introduction to elementary computer programming concepts. Emphasis is on techniques of problem analysis and the development of algorithms and programs. There are two versions of the course. Both provide adequate preparation for COM S/ENGRD 211. Both versions are not offered every semester.

COM S 100M Introduction to Computer Programming
Corequisite: MATH 111, 191, or equivalent.
This version starts with a seven-week introduction to programming in MATLAB. Iteration, functions, and arrays are introduced. During the second seven weeks of the course students examine how these ideas are handled in the object-oriented framework provided by the Java programming language. Throughout the course, examples and assignments are chosen to give the student an appreciation for computational science and engineering. The pace of the course assumes that the student has no prior programming experience.

COM S 100J Introduction to Computer Programming
This course is an introduction to programming using the Java programming language. A two-week unit on MATLAB is included. Topics include algorithms, language concepts, selection, repetition, functions, objects and classes, arrays, strings, and inheritance. Principles of software development and style are emphasized. The course assumes basic high school mathematics (no calculus) but no programming experience.

COM S 101 Introduction to Cognitive Science (also COGST 101, LING 170, and PSYCH 102)
Fall, summer. 3 credits.
For description, see COGST 101.

COM S 113 Introduction to C
Fall, spring. 1 credit. Usually weeks 1–4. Prerequisite: COM S 100 or equivalent programming experience. Credit is granted for both COM S 113 and 213 only if 113 is taken first. S-U grades only.
A brief introduction to the C programming language and standard libraries. Unix accounts are made available for students wishing to use that system for projects, but familiarity with Unix is not required. (Projects may be done using any modern implementation of C.) COM S 213 (C++ Programming) includes much of the material covered in 113. Students planning to take COM S 213 normally do not need to take 113.

COM S 114 Unix Tools
Fall. 1 credit. Usually weeks 5–8. Prerequisite: COM S 100 or equivalent programming experience. S-U grades only.
An introduction to Unix, emphasizing tools for file management, communication, process control, managing your Unix environment, and rudimentary shell scripts. Knowledge of at least one programming language is encouraged. Projects assume no previous knowledge of Unix or expertise in any particular language.

COM S 130 Introductory Web Programming (also CIS 130)
Fall, summer. 3 credits. No prerequisites.
Interactive online media such as the World Wide Web are revolutionizing the way we communicate. This course introduces students having little or no computer background to tools and techniques for creating interactive documents. Questions of both design and technical issues are emphasized. Students must think seriously about digital graphic impact and must learn how to do some relatively simple programming with a scripting language (such as JavaScript). Topics covered include HTML; JavaScript; interaction techniques (elementary DHTML); ways of coping with slow connections; the incorporation of sound, video, and images in web documents; animation techniques (Flash 5); ethics; and e-commerce.

COM S 172 Computation, Information, and Intelligence (also ENGRI 172)
Fall. 3 credits. Prerequisites: some knowledge of calculus.
An introduction to computer science using methods and examples from the field of artificial intelligence. Topics include game playing, search techniques, learning theory, compute-intensive methods, data mining, information retrieval, the web, natural language processing, machine translation, and the Turing test. This is not a programming course; rather, "pencil and paper" problem sets will be assigned. Not open to students who have completed the equivalent of COM S 100.

COM S 201 Cognitive Science in Context Laboratory (also COGST 201 and PSYCH 201)
Spring. 4 credits. Limited to 24 students. Prerequisite: concurrent or prior registration in Introduction to Cognitive Science (PSYCH 102/COGST 101/COM S 101/LING 170/PHIL 191) is suggested but not required. Knowledge of programming languages is not assumed. Fall, B. Halpern and staff; spring, D. Field and staff.
For description, see COGST 201.

COM S 211 Computers and Programming (also ENGRD 211)
Fall, spring, summer. 3 credits. Prerequisite: COM S 100 or an equivalent course in Java or C++.
Intermediate programming in a high-level language and introduction to computer science. Topics include program structure and organization, object-oriented programming (classes, objects, types, sub-typing), graphical user interfaces, algorithm analysis (asymptotic complexity, big "O" notation), recursion, data structures (lists, trees, stacks, queues, heaps, search trees, hash tables, graphs), simple graph algorithms. Java is the principal programming language.

COM S 212 Java Practicum
Fall, spring, summer. 1 credit. Letter grade only. Pre- or corequisite: COM S/ENGRD 211.
A project course that introduces students to the ways of software engineering using the Java programming language. The course requires the design and implementation of several large programs.

COM S 213 C++ Programming
Spring. 2 credits. Prerequisite: COM S 100 or equivalent programming experience. Students who plan to take COM S 113 and 213 must take 113 first. S-U grades only.
An intermediate-level introduction to the C++ programming language and the C/C++ standard libraries. Topics include basic statements, declarations, and types; stream I/O; user defined classes and types; derived classes, inheritance, and object-oriented programming; exceptions and templates. Recommended for students who plan to take advanced courses in computer science that require familiarity with C++ or C. Students planning to take COM S 213 normally do not need to take COM S 113; 213 includes most of the material taught in 113.

COM S 214 Advanced UNIX Programming and Tools
Spring. 1 credit. S-U grade only. Prerequisite: COM S 114 or equivalent.
A focus on Unix as a programming environment for people with a basic knowledge of Unix and experience programming in at least one language. Projects cover advanced shell scripts (sh, ksh, csh), Makefiles, programming and debugging tools for C and other languages, and more modern scripting languages such as Perl and Python. Students with little or no experience with Unix should take COM S 114 first.

COM S 215 Introduction to C#
Fall, spring. 1 credit. Prerequisite: COM S/ENGRD 211 or equivalent experience.
Introduces students to building applications in the .NET environment using the C# language.

COM S 230 Intermediate Web Design (also CIS 230)
Spring. 3 credits. Prerequisite: COM S 130 or equivalent.
Web programming requires the cooperation of two machines: the one in front of the viewer (client) and the one delivering the content (server). COM S 130 concentrates almost exclusively on the client side. The main emphasis in COM S 230 is learning about server side processing. Students begin by looking at interactions with databases, learning about querying both on paper and via SQL, and then, through a succession of projects, learn how to apply this understanding to the creation of an interactive data-driven site via the use of an integrated web site development tool such as ColdFusion. Also considered are techniques to enhance security, privacy, and reliability and ways of incorporating other programs. Toward the end of the course, students are shown how these development tools are working. Design issues are emphasized. A major component of the course is the creation of a substantial web site.

COM S 280 Discrete Structures
Fall, spring. 4 credits. Pre- or corequisite: COM S/ENGRD 211 or permission of instructor.
Covers mathematical aspects of programming and computing. Topics are chosen from the following: mathematical induction; logical proof; propositional and predicate calculus; combinatorics and discrete mathematics covering manipulation of sums, recurrence relations, and generating-function techniques; basic number theory; sets, functions, and relations; partially ordered sets; graphs; and algebraic structures.

COM S 312 Data Structures and Functional Programming
Fall, spring. 4 credits. Prerequisite: COM S 211/212 or equivalent programming experience. Should not be taken concurrently with COM S 314.
An advanced programming course that emphasizes functional programming techniques and data structures. Programming topics include recursive and higher-order procedures, models of programming language evaluation and compilation, type systems, and polymorphism. Data structures and algorithms covered include graph algorithms, balanced trees, memory heaps, and garbage collection. Also covered are techniques for analyzing program performance and correctness.

COM S 314 Computer Organization (also ECE 314)
Fall, spring. 4 credits. Prerequisite: COM S 211, COM S 312, or ENGRD 230 are recommended but not required. Should not be taken concurrently with COM S 312.
Basic computer organization. Topics include performance metrics, data formats, instruction sets, addressing modes, computer arithmetic, datapath design, memory hierarchies including caches and virtual memory, I/O devices, and bus-based I/O systems. Students learn assembly language programming and design a simple pipelined processor.

COM S 321 Numerical Methods in Computational Molecular Biology (also BIO BM 321 and ENGRD 321)
Fall. 3 credits. Prerequisites: at least one course in calculus, such as MATH 106, 111, or 191 and a course in linear algebra, such as MATH 221 or 294 or BTRY 417. COM S 100 or equivalent and some familiarity with iteration, arrays, and procedures.
An introduction to numerical computing using MATLAB organized around five applications: the analysis of protein shapes, dynamics, protein folding, score functions, and field equations. Students become adept at plotting, linear equation solving, least squares fitting, and cubic spline interpolation. More advanced problem-solving techniques that involve eigenvalue analysis, the solution of ordinary and partial differential equations, linear programming, and nonlinear minimization will also be treated. The goal of the course is to develop a practical computational expertise with MATLAB and to build mathematical intuition for the problems of molecular biology. COM S majors and minors may use only one of the following toward their degree: COM S 321, 322, or 421.

COM S 322 Introduction to Scientific Computation (also ENGRD 322)
Spring, summer. 3 credits. Prerequisites: COM S 100 and (MATH 222 or MATH 294).
An introduction to elementary numerical analysis and scientific computation. Topics include interpolation, quadrature, linear and nonlinear equation solving, least-squares fitting, and ordinary differential equations. The MATLAB computing environment is used. Vectorization, efficiency, reliability, and stability are stressed. Includes special lectures on parallel computation. COM S majors and minors may use only one of the following toward their degree: COM S 321, 322, or 421.

COM S 324 Computational Linguistics (also COGST 424, LING 424)
Fall, spring. 4 credits. Prerequisites: LING 203; labs involve work in the Unix environment; COM S 114 recommended.
For description, see LING 424.

COM S 330 Applied Database Systems (also CIS 330)
Spring. 3 credits. Prerequisite: COM S 211/ENGRD 211. COM S majors may use only one of the following toward their degree: CIS/COM S 330 or COM S 433.
An introduction to modern database systems and three-tier application development using database systems. Concepts covered include the relational model, query languages, data modeling, database tuning, three-tier architectures, Internet data formats and query languages, server- and client-side technologies, decision support systems, and an introduction to data mining. The course is targeted at users of database systems and at application development using database systems and includes several programming projects comprising the design and implementation of a database–backed web site.

COM S 381 Introduction to Theory of Computing
Fall, summer. 4 credits. Prerequisite: COM S 280 or permission of instructor. Credit will not be granted for both COM S 381 and COM S 481. Corrective transfers between COM S 381 and COM S 481 (in either direction) are encouraged during the first few weeks of instruction.
An introduction to the modern theory of computing: automata theory, formal languages, and effective computability.

COM S 411 Programming Languages
Fall. 4 credits. Prerequisite: COM S 312 or permission of instructor.
An introduction to the theory, design, and implementation of programming languages. Topics include operational semantics, type systems, higher-order function, scope, lambda calculus, laziness, exceptions, side effects, continuations, objects, and modules. Also discussed are logic programming, concurrency, and distributed and persistent programming.

COM S 412 Introduction to Compilers
Spring. 3 credits. Prerequisites: COM S 312 (or permission of instructor) and COM S 314. Corequisite: COM S 413.
An introduction to the specification and implementation of modern compilers. Topics covered include lexical scanning, parsing, type checking, code generation and translation, an introduction to optimization, and the implementation of modern programming languages. The course entails a substantial compiler implementation project.

COM S 413 Practicum in Compilers
Spring. 2 credits. Corequisite: COM S 412.
A compiler implementation project related to COM S 412.

COM S 414 Systems Programming and Operating Systems
Fall, spring, summer. 3 credits. Prerequisite: COM S 211, 212, 312 (or permission of instructor), and 314. Corequisite: COM S 415 in spring only.
An introduction to the logical design of systems programs, with emphasis on multiprogrammed operating systems. Topics include process synchronization, deadlock, memory management, input-output methods, information sharing, protection and security, and file systems. The impact of network and distributed computing environments on operating systems is also discussed.

COM S 415 Practicum in Operating Systems
Fall, spring. 2 credits. Corequisite: COM S 414.
The practical aspects of operating systems are studied through the design and implementation of an operating system kernel that supports multiprogramming, virtual memory, and various input-output devices. All the programming for the project is in a high-level language.

COM S 421 Numerical Analysis
Fall. 4 credits. Prerequisites: MATH 294 or equivalent, one additional mathematics course numbered 300 or above, and knowledge of programming.
Modern algorithms for systems of linear equations, systems of nonlinear equations, numerical optimization, and numerical solution of differential equations. Some discussion of methods suitable for parallel computation. This course requires more mathematical sophistication than COM S 322. COM S majors and minors may use only one of the following toward their degree: COM S 321, 322 or 421.

COM S 426 Introduction to Computational Biology
Fall. 3 credits. Prerequisites: COM S/ENGRD 211, COM S 280.
Computational biology is a relatively new field that is rooted in two different disciplines: computer science and molecular biology. It is concerned with the study of biological systems and phenomena, in search for explanations, rules, patterns, and regularities. The focus of this course is the set of algorithms, tools, and models used today to analyze biological data and recover and discover hidden information. These tools can be used to predict the function of new genes, discover hidden motifs that are biologically significant, study evolutionary processes, better understand cellular "computations," and aid in the development of new therapeutic agents for treatment of various diseases. Some of the topics covered are sequence analysis (alignment, multiple sequence alignment), motif detection, phylogenetic trees, secondary structure prediction, Hidden Markov Models, and analysis of gene expression data.

This course is intended mostly for students in the computational sciences because it focuses on algorithms and mathematical models. No knowledge of biology is needed (the course starts with a brief overview of the basic entities and the central dogma of molecular biology, and other relevant terms are introduced as needed).

COM S 427 Practicum in Computational Biology
Fall. 2 credits. Pre- or corequisite: COM S 426.
In this course, students develop a system or an application to analyze biological data. Possible applications are a database system to manipulate multiple data types, a learning system to detect hidden patterns in massive biological data sets, a software suite for sequence comparison or analysis of gene expression data, etc.

COM S 430 Information Discovery (also CIS 430)
Fall. 3 credits. Prerequisite: COM S 211 or equivalent.
This course studies the methods used to search for and discover information in large-scale systems. The emphasis is on information retrieval applied to textual materials, but there is some discussion of other formats. The course includes techniques for searching, browsing, and filtering information and the use of classification systems and thesauruses. The techniques are illustrated with examples from web searching and digital libraries.

COM S 431 Web Information Systems (also CIS 431; formerly CIS/COM S 502)
Spring. 3 credits. Prerequisites: COM S 211 and some familiarity with the technology of web sites.
This course examines the architecture of web information systems such as distributed digital libraries and electronic publishing systems. Many of the topics presented in the course are the subject of current research and development at Cornell, other universities, and in standards organizations such as the World Wide Web Consortium. Course content mixes exploration of current tools for building web information systems such XML, XSLT, and RDF with broader concepts such as techniques for knowledge representation and description, object models for content representation, and legal and economic impacts of web information. A theme that runs throughout the course is the relationship between traditional information environments, exemplified by libraries, and the distributed information environment of the web.

COM S 432 Introduction to Database Systems
Fall. 3 credits. Prerequisites: COM S 312, or 211/212, and permission of instructor. Recommended: COM S 213 and strong programming skills in C or C++.
An introduction to modern relational database systems. Concepts covered include storage structures, access methods, query languages, query processing and optimization, transaction processing, and database design theory. The course primarily covers the internals of database systems and includes four large programming assignments in C++.

COM S 433 Practicum in Database Systems
Fall. 2 credits. Corequisite: COM S 432. COM S majors may use only one of the following toward their degree: CIS/COM S 330 or COM S 433.
An introduction to building web-database applications. Students implement a small e-commerce system using Active Server Pages, Java Server Pages, Cookies, and Servlets. The practicum also introduces technologies such as XML/XPath/XSLT and WAP. COM S majors may use only one of the following toward their degree: CIS/COM S 333 or COM S 433.

COM S 465 Computer Graphics I (also ARCH 374)
Fall. 3 credits. Prerequisite: COM S/ENGRD 211. May not be taken after completion of COM S 417.
An introduction to the principles of computer graphics in two and three dimensions. Topics include human visual perception, digital images, rasterization and anti-aliasing, 2D and 3D affine geometry, perspective and 3D viewing, spline curves, elementary 3D surface modeling, and ray tracing. Homework assignments require some programming.

COM S 467 Computer Graphics II
Spring. 3 credits. Prerequisite: COM S 465.
This course covers the principles of computer graphics in detail. A wide variety of topics are covered, including 3D transformations, the hardware graphics pipeline, advanced texturing and shading, visual perception and color science, rendering algorithms including global illumination, animation, user interfaces, visualization, and 3D surface modeling.

COM S 468 Computer Graphics Practicum
Spring. 2 credits. Prerequisite: COM S 465. Corequisite: COM S 467.
This course provides COM S 467 students with hands-on experience in computer graphics programming. Programming assignments cover 3D transformations, modeling, shading, rendering, animation, and user interfaces. The course uses Java, OpenGL, and Cg for code development.

COM S 472 Foundations of Artificial Intelligence
Fall. 3 credits. Prerequisites: COM S/ENGRD 211 and COM S 280 (or equivalent).
A challenging introduction to the major subareas and current research directions in artificial intelligence. Topics include knowledge representation, heuristic search, problem solving, natural-language processing, game-playing, logic and deduction, planning, and machine learning.

COM S 473 Practicum in Artificial Intelligence
Fall. 2 credits. Corequisite: COM S 472.
Project portion of COM S 472. Topics include knowledge representation systems, search procedures, game-playing, automated reasoning, concept learning, reinforcement learning, neural nets, genetic algorithms, planning, and truth maintenance.

[COM S 474 Introduction to Natural Language Processing (also COGST 474, LING 474)
Fall. 4 credits. Prerequisites: COM S 211. Not offered every year.]

COM S 478 Machine Learning
Spring. 4 credits. Prerequisites: COM S 280, 312, and basic knowledge of linear algebra and probability theory.
Learning and classifying are two of our basic abilities. Machine learning is concerned with the question of how to train computers to learn from experience, to adapt and make decisions accordingly. This course introduces the set of techniques and algorithms that constitute machine learning as of today, including inductive inference of decision trees, the parametric-based Bayesian learning approach, Bayesian belief networks and Hidden Markov Models, non-parametric methods, discriminent functions and support vector machines, neural networks, stochastic methods such as genetic algorithms, unsupervised learning and clustering, and other issues in the theory of machine learning. These techniques are used today to automate procedures that were previously performed by humans as well as to explore untouched domains of science.

COM S 481 Introduction to Theory of Computing
Fall. 4 credits. Prerequisite: COM S 280 or permission of instructor. Credit will not be granted for both COM S 381 and 481.

Corrective transfers between COM S 481 and 381 (in either direction) are encouraged during the first few weeks of instruction.

A faster-moving and deeper version of COM S 381.

COM S 482 Introduction to Analysis of Algorithms

Spring, summer. 4 credits. Prerequisites: COM S 280, 312, and either 381 or 481, or permission of instructor.

Techniques used in the creation and analysis of algorithms. Combinatorial algorithms, computational complexity, NP-completeness, and intractable problems.

COM S 483 Quantum Computation (also PHYS 481 and 681)

Spring. 2 credits. Prerequisite: familiarity with the theory of vector spaces over the complex numbers. Not offered every year.

For description, see PHYS 481.

COM S 486 Applied Logic (also MATH 486)

Fall or spring. 4 credits. Prerequisites: MATH 222 or 294, COM S 280 or equivalent (such as MATH 332, 432, 434, 481), and some additional course in mathematics or theoretical computer science.

Propositional and predicate logic, compactness and completeness by tableaux, natural deduction, and resolution. Equational logic. Herbrand Universes and unification. Rewrite rules and equational logic, Knuth-Bendix method, and the congruence-closure algorithm and lambda-calculus reduction strategies. Topics in Prolog, LISP, ML, or Nuprl. Applications to expert systems and program verification.

COM S 490 Independent Reading and Research

Fall, spring. 1–4 credits.

Independent reading and research for undergraduates.

COM S 501 Software Engineering

Spring. 4 credits. Prerequisite: COM S 211 or equivalent experience programming in Java or C++.

An introduction to the practical problems of specifying, designing, and building large, reliable software systems. Students work in teams on projects for real clients. This work includes a feasibility study, requirements analysis, object-oriented design, implementation, testing, and delivery to the client. Additional topics covered in lectures include professionalism, project management, and the legal framework for software development.

COM S 504 Applied Systems Engineering I (also CEE 504, ECE 512, M&AE 591, OR&IE 512, SYSEN 510)

Fall. 3 credits. Prerequisites: senior or graduate standing in an engineering field; concurrent or recent (past two years) enrollment in a group-based project with a strong system design component that is approved by a course instructor.

For description, see M&AE 591.

COM S 505 Applied Systems Engineering II (also CEE 505, ECE 513, M&AE 592, OR&IE 513, SYSEN 520)

Spring. 3 credits. Prerequisite: Applied System Engineering I (CEE 504, COM S 504, ECE 512, M&AE 591, OR&IE 512).

For description, see M&AE 592.

COM S 513 System Security

Spring. 4 credits. Prerequisites: COM S 414 or 519 and familiarity with JAVA programming language.

This course discusses security and survivability for computers and communications networks. The course includes discussions of policy issues (e.g., the national debates on cryptography policy) as well as the discussions of the technical alternatives for implementing the properties that comprise "trustworthiness" in a computing system. Mechanisms for authorization and authentication as well as cryptographic protocols are covered.

COM S 514 Intermediate Computer Systems

Fall. 4 credits. Prerequisites: COM S 414 or permission of instructor.

This course focuses on practical issues in designing and implementing distributed software. Topics vary depending upon instructor. Recent offerings have covered object-oriented software development methodologies and tools, distributed computing, fault-tolerant systems, and network operating systems or databases. Students undertake a substantial software project. Many students obtain additional project credit by coregistering in COM S 490, 515, or 790.

COM S 519 Computer Networks

Spring. 4 credits. Prerequisites: COM S 314 or permission of instructor. Not offered every year.

Introduction to computer networks with an emphasis on evolving Internet standards. Detailed introduction to networking protocols for reliable data transfer, flow control, congestion control, naming and addressing, routing, error control, and multiple access. Fundamentals of layered protocols and techniques for protocol design and implementation. The course is project-oriented and requires substantial programming experience.

COM S 522 Computational Tools and Methods for Finance

Spring. 4 credits. Prerequisites: programming experience (e.g., C, FORTRAN, or MATLAB), some knowledge of numerical methods, especially numerical linear algebra. Not offered every year.

This course provides a hands-on introduction to computational methods and tools used in finance. Students study both the underlying methods and efficient implementation. The MATLAB Financial Toolbox, along with additional MATLAB tools, are used extensively. The underlying numerical techniques discussed include nonlinear least-squares procedures (regression), basic linear algebra, linear and nonlinear optimization, finite-difference methods for PDEs, quadratic programming (and linear complementarity problems), and specialized tree (and lattice) evaluation methods.

COM S 572 Heuristic Methods for Optimization (also CEE 509)

Spring. 3 or 4 credits. Prerequisites: COM S/ENGRD 211 or 322 or CEE/ENGRD 241, or graduate standing, or permission of instructor. Not offered every year.

This course describes a variety of heuristic search methods including simulated annealing, tabu search, genetic algorithms, derandomized evolution strategy, random walk, and direct search algorithms. Algorithms are used to find values of discrete and/or continuous variables arising in optimization and model fitting. Applications are discussed in a range of areas including some of the following: artificial intelligence, scheduling, protein folding, economic planning, water quality protection, telecommunications, and robotics. The advantages and disadvantages of heuristic search methods for both serial and parallel computation are discussed in comparison to other optimization algorithms.

[COM S 574 Language Technologies

Fall. 3 credits. Prerequisites: COM S 472 or 478 or 578 or the equivalent. Not offered every year.

This course studies computational techniques for large-scale text-processing applications, including information retrieval, text classification, information extraction, document clustering, document ranking, summarization, topic detection and tracking, and question answering. The course focuses on statistical and machine learning approaches to these natural language processing tasks as well as methods for their empirical evaluation.]

COM S 578 Empirical Methods in Machine Learning and Data Mining

Fall. 4 credits. Prerequisites: COM S 280 and 312 or equivalent.

This implementation-oriented course presents a broad introduction to current algorithms and approaches in machine learning, knowledge discovery, and data mining and their application to real-world learning and decision-making tasks. The course also covers experimental methods for comparing learning algorithms, for understanding and explaining their differences, and for exploring the conditions under which each is most appropriate.

[COM S 601 System Concepts

Fall. 3 credits. Prerequisites: open to students enrolled in the COM S Ph.D. program. Not offered every year.]

COM S 611 Advanced Programming Languages

Fall. 4 credits. Graduate standing or permission of instructor.

A study of programming paradigms: functional, imperative, concurrent, and logic programming. Models of programming languages, including the lambda calculus. Type systems, polymorphism, modules, and other object-oriented constructs. Program transformations, programming logic, and applications to programming methodology.

COM S 612 Compiler Design for High-Performance Architectures

Spring. 4 credits. Prerequisites: COM S 314 and 412 or permission of instructor.

Compiler design for pipelined and parallel architectures. Program analysis: data and control dependencies, dataflow analysis, efficient solution of dataflow equations, dependence tests, solution of Diophantine equations. Architecture and code generation for instruction-level parallel (ILP) processors: pipelined, VLIW and superscalar architectures, code reorganization and software pipelining. Architecture and code generation for multi-processors: shared- and distributed-memory architectures, latency tolerance and avoidance, loop transformations to enhance parallelism and locality of reference.

[COM S 613 Concurrent Programming
Spring. 4 credits. Prerequisite: COM S 414 or permission of instructor. Not offered every year.]

COM S 614 Advanced Systems
Spring. 4 credits. Prerequisite: COM S 414 or permission of instructor.
An advanced course in systems, emphasizing contemporary research in distributed systems. Topics may include communication protocols, consistency in distributed systems, fault-tolerance, knowledge and knowledge-based protocols, performance, scheduling, concurrency control, and authentication and security issues.

COM S 615 Adaptive Systems
Fall. 4 credits. Prerequisites: COM S 614 recommended.
This course broadly examines self-organization in distributed systems, with particular emphasis on peer-to-peer and mobile ad hoc networks. Students read recent research papers on proactive, reactive and hybrid routing protocols for ad hoc networks, system support for mobility, peer-to-peer systems and their applications.

COM S 621 Matrix Computations
Fall. 4 credits. Prerequisites: MATH 411 and 431 or permission of instructor.
Stable and efficient algorithms for linear equations, least squares, and eigenvalue problems. Direct and iterative methods are considered. The MATLAB system is used extensively.

[COM S 622 Numerical Optimization and Nonlinear Algebraic Equations
Spring. 4 credits. Prerequisite: COM S 621. Offered odd-numbered years only.
Modern algorithms for the numerical solution of multidimensional optimization problems and simultaneous nonlinear algebraic equations. Emphasis is on efficient, stable, and reliable numerical techniques with strong global convergence properties: quasi-Newton methods, modified Newton algorithms, and trust-region procedures. Special topics may include large-scale optimization, quadratic programming, and numerical approximation.]

COM S 624 Numerical Solution of Differential Equations
Spring. 4 credits. Prerequisites: previous exposure to numerical analysis (e.g., COM S 421 or 621) and differential equations, and knowledge of MATLAB. Offered in even-numbered years.
Finite difference methods for the solution of ordinary and partial differential equations. A fast-moving course that begins with a three-week survey of numerical methods for ODEs, then moves on to Fourier analysis and methods for PDEs, especially parabolic and hyperbolic equations. Other topics covered include numerical stability, finite element methods, Hamiltonian problems, and computational issues such as mesh generation and sparse matrix computation for PDES.

COM S 626 Computational Molecular Biology
Spring. 4 credits. Prerequisites: familiarity with linear programming, numerical solutions of ordinary differential equations, and nonlinear optimization methods.
Problems and algorithms in computational molecular biology. Topics include sequences (alignment, scoring functions, complexity of searches and alignment, secondary structure prediction, families, and function), the protein folding problem (lattice models, lattice searches, the HP model, chemical potentials, statistical potentials, funnels, complexity and model verification, global optimization, homology, threading), and the dynamics of complex biosystems (the Molecular Dynamics method, long-range forces, statistics of flexible systems, reduced models).

COM S 627 Computational Biology: The Machine Learning Approach
Spring. 4 credits. Prerequisites: COM S 426 or 626 and COM S 478 or 578 or permission of instructor.
This is a graduate-level course in computational biology that focuses on machine learning models and their application to computational problems in biology. Some topics covered are supervised (Support Vector Machines, Hidden Markov Models, deterministic and probabilistic suffix trees) and unsupervised (embedding, PCA, ICA, clustering) learning in computational biology, advanced statistical analysis of sequences, analysis of microarrays, and modeling of complex systems (Bayesian Belief Networks, DEA).

COM S 630 Representing and Accessing Digital Information (also CIS 630)
Fall. 4 credits. Prerequisites: COM S 472 or 478 or 578 or the equivalent.
This course covers the representation, organization, and access of digital information with an emphasis on textual information. Topics include structured and semistructured data, information retrieval, natural language processing, and machine learning, with links to work in databases, data mining, and computational linguistics.

COM S 632 Advanced Database Systems
Spring. 4 credits. Prerequisite: COM S 432/433 or permission of instructor.
A variety of advanced issues ranging from transaction management to query processing to data mining. Involves extensive paper reading and discussion. Development of a term project with research content is required.

COM S 664 Machine Vision
Fall. 4 credits. Prerequisites: undergraduate-level understanding of algorithms and MATH 221 or equivalent.
An introduction to computer vision, with an emphasis on discrete optimization algorithms and on applications in medical imaging. The following topics are covered: edge detection, image segmentation, stereopsis, motion and optical flow, active contours, and the Hausdorff distance. Students are required to implement several of the algorithms covered in the course and complete a final project.

COM S 665 Advanced Rendering
Fall or spring. 4 credits. Prerequisites: COM S 465 and 467 or equivalent and an undergraduate-level understanding of algorithms, probability and statistics, vector calculus, and programming. Offered fall 2003.
This course covers advanced topics in realistic rendering with a focus on fast/interactive techniques. Topics include light transport and global illumination, Monte-Carlo ray tracing and radiosity, hardware rendering, and image-based rendering.

COM S 667 Physically Based Rendering
Fall or spring. 4 credits. Prerequisites: COM S 465 and 467 or equivalent and an undergraduate-level understanding of algorithms, programming, and vector calculus. Offered spring 2004.
An advanced course in realistic image synthesis, focusing on the computation of physically accurate images. Topics include radiometry; light transport and global illumination; rendering with participating media; advanced models for material properties; and physical measurement of light sources, images, and materials.

COM S 671 Introduction to Automated Reasoning
Fall. 4 credits. Prerequisite: (COM S 611 and graduate standing) or permission of instructor.
Topics in modern logic needed to understand and use automated reasoning systems such as HOL, Nuprl, and PVS. Special emphasis is on type theory and logic and on tactic-oriented theorem proving.

COM S 672 Advanced Artificial Intelligence
Spring. 4 credits. Prerequisites: COM S 472 or permission of instructor.
Artificial intelligence (AI) provides many computational challenges. This course covers a variety of areas in AI, including knowledge representation, automated reasoning, learning, game-playing, and planning, with an emphasis on computational issues. Specific topics include stochastic reasoning and search procedures, properties of problem encodings, issues of syntax and semantics in knowledge representation, constraint satisfaction methods and search procedures, and critically constrained problems and their relation to phase-transition phenomena. In addition, connections between artificial intelligence and other fields, such as statistical physics, operations research, and cognitive science are explored.

COM S 674 Natural Language Processing
Spring. 3 credits. Prerequisites: COM S 472 or permission of instructor. COM S 474 is NOT a prerequisite. Not offered every year.
This course presents a graduate-level introduction to natural language processing, the primary concern of which is the study of human language use from a computational perspective. The course covers syntactic analysis, semantic interpretation, and discourse processing, examining both symbolic and statistical approaches. Possible topics include information extraction, natural language generation, memory models, ambiguity resolution, finite-state methods, mildly context-sensitive formalisms, deductive approaches to interpretation, machine translation, and machine learning of natural language.

COM S 676 Reasoning about Knowledge
Fall. 4 credits. Prerequisites: mathematical maturity and an acquaintance with propositional logic.
Knowledge plays a crucial role in distributed systems, game theory, and artificial intelligence. Material examines formalizing reasoning about knowledge and the extent to which knowledge is applicable to those areas. Issues include common knowledge, knowledge-based programs, applying knowledge to analyzing distributed systems, attainable states of knowledge, modeling resource-bounded reasoning, and connections to game theory.

[COM S 677 Reasoning about Uncertainty
Fall. 4 credits. Prerequisites: mathematical maturity and an acquaintance with propositional logic. Not offered 2003–2004.
Examines formalizing reasoning about and representing uncertainty, using formal logical approaches as a basis. Topics: logics of probability, combining knowledge and probability, probability and adversaries, conditional logics of normality, Bayesian networks, qualitative approaches to uncertainty, going from statistical information to degrees of belief, and decision theory.]

COM S 678 Advanced Topics in Machine Learning
Spring. 4 credits. Prerequisites: COM S 478 or equivalent, or COM S 578 or equivalent, or permission of instructor.
This course extends and complements COM S 478 and COM S 578, giving in-depth coverage of new and advanced methods in machine learning. In particular, we connect to open research questions in machine learning, giving starting points for future work. The content of the course reflects an equal balance between learning theory and practical machine learning, making an emphasis on approaches with practical relevance. Topics include support vector machines, clustering, Bayes nets, boosting, model selection, learning orderings, and inductive transfer.

COM S 681 Analysis of Algorithms
Fall. 4 credits. Prerequisite: COM S 482 or graduate standing.
Methodology for developing efficient algorithms, primarily for graph theoretic problems. Understanding of the inherent complexity of natural problems via polynomial-time algorithms, randomized algorithms, NP-completeness, and randomized reducibilities. Also covers topics such as parallel algorithms and efficient data structures.

COM S 682 Theory of Computing
Spring. 4 credits. Prerequisite: (COM S 381 or 481) and (COM S 482 or 681) or permission of instructor.
Advanced treatment of theory of computation, computational-complexity theory, and other topics in computing theory.

[COM S 683 Advanced Design and Analysis of Algorithms
Spring. 4 credits. Prerequisites: COM S 681 or permission of instructor. Not offered every year.]

[COM S 684 Approximation and Network Algorithms
Fall. 4 credits. Prerequisites: COM S 681 or permission of instructor. Not offered every year.]

COM S 685 The Structure of Information Networks (also CIS 685)
Spring. 4 credits. Prerequisite: COM S 482.
Information networks such as the World Wide Web are characterized by the interplay between heterogeneous content and a complex underlying link structure. This course covers recent research on algorithms for analyzing such networks and models that abstract their basic properties. Topics include combinatorial and probabilistic techniques for link analysis, centralized and decentralized search algorithms, generative models for networks, and connections with work in the areas of social networks and citation analysis.

[COM S 686 Logics of Programs
Spring. 4 credits. Prerequisites: COM S 481, 682, and MATH 481 or MATH/COM S 486. Not offered every year.]

COM S 709 Computer Science Colloquium
Fall, spring. 1 credit. S-U grades only. For staff, visitors, and graduate students interested in computer science.
A weekly meeting for the discussion and study of important topics in the field.

COM S 711 Seminar in Advanced Programming Languages
Fall, spring. 3 credits.

COM S 713 Seminar in Systems and Methodology
Fall, spring. 4 credits. Prerequisites: a graduate course employing formal reasoning such as COM S 611, 613, 671, a logic course, or permission of instructor. Not offered every year.
Discussion of contemporary issues in the design and analysis of computing systems. Emphasis is on the proper use of rigor, models, and formalism.

COM S 715 Seminar in Programming Refinement Logics
Fall, spring. 4 credits. Prerequisite: permission of instructor.
Topics in programming logics, possibly including type theory, constructive logic, decision procedures, heuristic methods, extraction of code from proofs, and the design of proof-development and problem-solving systems.

COM S 717 Topics in Parallel Architectures
Fall. 4 credits. Prerequisite: COM S 612 or permission of instructor. Not offered every year.
Covers topics in parallel computers. Material includes: architectures of parallel computers, parallelizing compilers, operating systems for parallel computers, and languages (functional and logic-programming languages) designed for parallel computation.

COM S 719 Seminar in Programming Languages
Fall, spring. 4 credits. Prerequisite: COM S 611 or permission of instructor. S-U grades only.

COM S 721 Topics in Numerical Analysis
Fall, spring. 4 credits. Prerequisite: COM S 621 or 622 or permission of instructor. Not offered every year; semester TBA.
Topics are chosen at instructor's discretion.

COM S 726 Problems and Perspectives in Computational Molecular Biology (also PL BR 726)
Fall, spring. 1 credit. S-U grades only.
This is a weekly seminar series discussing timely topics in computational molecular biology. The course addresses methodological approaches to sequence and structure analysis, function prediction, study of evolutionary relationships, and analysis of large biological systems. Statistical and deterministic computational approaches are covered, and specific and detailed biological examples are discussed. In each topic, we select one or two representative papers that made significant advances in this field. The lectures are given by faculty and students. The seminar is open to all from the life sciences, computational sciences, and physical sciences. We try to bridge these disciplines by pairing students/faculty from complementary backgrounds.

COM S 732 Seminar in Database Systems
Fall, spring. 4 credits. S-U grades only.

[COM S 750 Evolutionary Computation and Design Automation (also CIS 750, M&AE 650)
Fall. 4 credits. Prerequisite: programming experience or permission of instructor. Not offered every year.]

[COM S 751 Media Research and Critical Design (also CIS 751)
Fall. 4 credits. Prerequisites: graduate standing in COM S or equivalent ability to read technical research papers. Contact instructor if unsure of qualifications. Not offered every year.]

[COM S 752 Seminar on Scholarly Information Architecture (also CIS 752)
Fall. 3 credits. Prerequisite: concurrent enrollment in COM S 502 or equivalent experience. S-U grades only. Not offered every year.]

COM S 754 Systems Research Seminar
Fall, spring. 1 credit. S-U grades only.

COM S 772 Seminar in Artificial Intelligence
Fall, spring. 4 credits. Prerequisites: permission of instructor. S-U grades only.

COM S 775 Seminar in Natural Language Understanding
Fall, spring. 2 credits.
Informal weekly seminar in which current topics in natural language understanding and computational linguistics are discussed.

COM S 786 Introduction to Kleene Algebra
Spring. 4 credits. Prerequisites: COM S 481 required; COM S 482 or 681, COM S 682, elementary logic (MATH 481 or 681), algebra (MATH 432) recommended.
Kleene algebra is an algebraic system that axiomatically captures the properties of a natural class of structures arising in logic and computer science. It has appeared in various guises in relational algebra, semantics and logics of programs, automata and formal language theory, and the design and analysis of algorithms. In this course, we review the history of the development of Kleene algebra and Kleene algebra with tests (Kleene/Boolean algebra). We study models, compare axiomatizations, and derive completeness, expressiveness, and complexity results. We also discuss various applications in program schematology, program verification, compiler optimization, and programming language semantics and logic.

COM S 789 Seminar in Theory of Algorithms and Computing
Fall, spring. 4 credits. Prerequisite: permission of instructor. S-U grades only.

COM S 790 Special Investigations in Computer Science
Fall, spring. Prerequisite: permission of a computer science adviser. Letter grade only.
Independent research or Master of Engineering project.

COM S 990 Special Investigations in Computer Science
Fall, spring. Prerequisite: permission of a computer science adviser. S-U grades only.
Doctoral research.